VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Rozhraní pro správu routeru na platformě Mikrotik

System for Management of Mikrotik Route

2014 Roman Bednář

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Zadání bakalářské práce

Student: **Roman Bednář**

Studijní program: B2647 Informační a komunikační technologie

Studijní obor: 2612R025 Informatika a výpočetní technika

Téma: Rozhraní pro správu routeru na platformě Mikrotik
System for Management of Mikrotik Router

Zásady pro vypracování:

Vytvořte sytém obsahující uživatelské a konfigurační rozhraní. Systém bude určen pro správu síťových zařízení provozovaných na platformě Mikrotik. Systém bude komunikovat pomocí API, bude zabezpečen uživatelskými právy a musí zabezpečit ochranu vůči chybám ze strany nezkušeného uživatele. Rozhraní umožní blokovat provoz pro konkrétní IP adresy, tak i pro definovaný rozsah adres, zadávat blokované stránky či klíčová slova, časové vypínání, zobrazení počitadla úspěšných zablokování stránek, zobrazení celkové velikosti internetového provozu.

1. Popište platformu Mikrotik a používané API.
2. Definujte požadavky na uživatelskou a konfigurační část systému.
3. Navrhněte vhodné prostředí pro vývoj aplikace.
4. Systém realizujte a proveďte testování v reálném provozu.
5. Vytvořte uživatelskou příručku.

Seznam doporučené odborné literatury:

[1] STREBE, Matthew a Charles PERKINS. Firewally a proxy-servery. Vyd. 1. Brno: Computer Press, 2003, xxi, 450 s. ISBN 80-722-6983-6
[2] MACDONALD, Matthew, Adam FREEMAN a Mario SZPUSZTA. ASP.NET 4 a C# 2010: tvorba dynamických stránek profesionálně. Vyd. 1. Překlad Jan Pokorný. Brno: Zoner Press, 2011, 880 s. Encyklopedie Zoner Press. ISBN 978-80-7413-131-8
[3] Manual: Mikrotik API. [online]. [cit. 2013-09-31]. Dostupné z: http://wiki.mikrotik.com/wiki/Manual:API
[4] Manual: Mikrotik. [online]. [cit. 2013-09-31]. Dostupné z: http://wiki.mikrotik.com/wiki/

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. David Seidl, Ph.D.**

Datum zadání: 01.09.2013

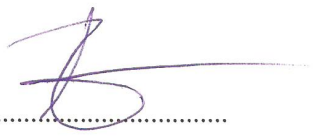
Datum odevzdání: 07.05.2014

doc. Dr. Ing. Eduard Sojka
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

„Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal."

V Havířově 30.července 2014



.....

Rád bych touto cestou poděkoval Ing. Davidu Seidlovi, Ph.D. za odbornou pomoc, konzultace, užitečné rady a připomínky při vytváření bakalářské práce.

Abstrakt

Cílem mé práce je navrhnout a implementovat webovou aplikaci, sloužící ke správě internetového provozu v počítačových učebnách, na stávající síťové architektuře, jehož hlavní částí je směrovač na platformě Mikrotik. Při návrhu byla zvážena možnost dalšího rozšíření aplikace, proto byl brán zřetel na modulárnost celého systému. K implementaci byla použita architektura ASP.NET MVC 4 a programovací jazyk C#. Ke komunikaci s RouterOS se využívá rozhraní Mikrotik API, jakož to nástroj pro tvorbu vlastního řešení správy RouterOS. Aplikace slouží svými funkcemi jako náhrada komerčních řešení.

Klíčová slova: Webová aplikace, Mikrotik, Mikrotik API, RouterOS, ASP.NET, MVC, C#, správa internetu

Abstract

The aim of my thesis is to design and implement web application for managing Internet in computer classrooms, on current network architecture, whose main component is a router on platform Mikrotik. During designing the possibility of additional extension the application was considered, therefore the modularity of the system was taken into account . The architecture ASP, NET MVC 4 and programming language C# were used for implement . To communicate with the RouterOS interface Mikrotik API is used, as a tool for creating custom management solutions RouterOS. The application serves its functions as a replacement of commercial solutions.

Keywords: Web applications,Mikrotik, Mikrotik API, RouterOS, ASP.NET, MVC, C#, governance of the Internet

Seznam použitých zkratek a symbolů

ADSL	–	Asymmetric Digital Subscriber Line
API	–	Application Programming Interface
BFD	–	Bidirectional Forwarding Detection
BGP	–	BorderGateway Protocol
CPU	–	Central Processing Unit
DHCP	–	Dynamic Host Configuration Protocol
DNS	–	Domain Name System
DSCP	–	Differentiated Services Code Point
EoIP	–	Ethernet over IP
FTP	–	File Transfer Protocol
HTTP	–	Hypertext Transfer Protocol
HW	–	Hardware
IANA	–	Internet Assigned Numbers Authority
ICANN	–	Internet Corporation for Assingned Names and Numbers
ICMP	–	Internet Control Message Protocol
IGMP	–	Internet Group Management Protokol
IOS	–	Internetwork Operating System
IP	–	Internet Protocol
IPIP	–	International Personality Item Pool
IPS	–	Intrusion Prevention System
IPSec	–	Internet Protocol Security
IPv6	–	Internet Protocol version 6
ISP	–	Internet service provider
LAN	–	Local Area Network
L2TP	–	Layer 2 Tunneling Protocol
md5	–	Message-Digest verze 5
NAT	–	Network Adress Translation
NTP	–	Network Time Protocol
PCQ	–	Per Connection Queuing
OS	–	Operating system
OSPF	–	Open Shortest Path First
PPPoE	–	Point-to-Point Protocol over Ethernet
PPTP	–	Point-to-Point Tunneling Protocol

RAID	–	Redundant Array of Independent Disks
RAM	–	Random - access memory
RED	–	Random Early Drop
RIP	–	Routing Information Protocol
SFQ	–	Stochastic Fairness Queuing
SOCKS	–	Socket Secure
SSH	–	Secure Shell
SSTP	–	Secure Socket Tunneling Protocol
SW	–	Software
ToS	–	Type of Service
TTL	–	Time to live
UDP	–	User Datagram Protocol
URL	–	Uniform Resource Locator
USB	–	Universal Serial Bus
VPN	–	Virtual Private Network
Wi-Fi	–	Wireless Fidelity

Obsah

Seznam tabulek

Seznam obrázků

Seznam výpisů zdrojového kódu

1 Úvod

Internet jako dobrý sluha, ale špatný pán. Tak by se dala popsat kapitola internet v životě dítěte. Internet je neomezený zdroj informací, zábavy, ale taky plný nástrah a hrozeb.

„Internet jako celek nikdo nevlastní ani přímo neřídí. Samozřejmě je nutné Internet nějak koordinovat, aby nenastala úplná anarchie. Pro tyto účely existuje několik nadnárodních organizací, např. ICANN či IANA, které různé věci centrálně evidují, koordinují, vymýšlí a zavádí standardy apod. Cílem je, aby síť byla zejména oproštěna od politických vlivů, což bohužel není zcela možné.“ [12]

V dnešní době je internet jako médium dostupný skoro na každém kroku. V podobě Wi-Fi Hotspotů se s internetem setkáváme v restauracích, nákupních centrech, na nádražích, na letištích, na benzínových pumpách a v místech, kde lidé tráví čas. Čím dál tím častěji se s internetem setkáváme v mobilních zařízeních. V neposlední řadě pak ve škole v rámci výuky.

„Dnešní mladá generace tráví s médii více času než jakoukoli jinou aktivitou vyjma spánku. Děti a dospívající užívají média stále intenzivněji, neboť často konzumují více médií najednou. Ačkoli platí, že nejvíce času tráví mladí sledováním televize, ve starších věkových skupinách hraje již důležitější roli internet. Děti a dospívající používají internet nejčastěji ke školní práci, hraní her, sledování videoklipů a ke komunikování po chatu a na sociálních sítích. Existují různé styly, jimiž děti a dospívající internet užívají. Některé děti se specializují na hraní her a sledování videoklipů, jiné jsou aktivní v kreativních činnostech (přeposílají a sdílejí zprávy, vytvářejí blogy) a jiné zase v navštěvování sociálních sítí.

Život dětí a dospívajících v kyberprostoru sebou nese řadu pozitiv i potencionálních rizik. Online prostředí umožňuje mladé generaci poznávat svět a učit se, pomáhá jim hledat vlastní identitu, navazovat nové vztahy a zařazovat se do sociálních skupin. Rizika užívání internetu a dalších nových médií tkví například v jejich možném zneužití s cílem ublížit druhým osobám, pronásledovat je nebo s nimi manipulovat.“ [11]

Proto je důležité děti seznamovat s internetem již od začátku školní docházky. Vymezit hranice užívání internetu, tak aby neovlivňoval negativně vývoj dítěte. To je nelehký úkol pro vyučujícího kantora, seznámit žáky s možnostmi využívání internetu, neboť získat si jejich pozornost a uhlídat, aby internet používali pro účely výuky a ne pro své potěšení či zábavu, je opravdu těžké. A to je důvod, proč jsem si zvolil toto téma bakalářské práce. Jelikož se v rámci náplně své práce často setkávám s kantory, kteří razí novou formu multimediální výuky, tak často řeší nekázeň a neochotu se věnovat výuce. Touto cestou bych rád usnadnil práci kantorů, tím že se jim pokusím poskytnout nástroj pro správu internetového provozu na počítačové učebně v době výuky.

2 Rozbor problému

Řízením internetového provozu na počítačové učebně se rozumí, umožnit kantorovi úplnou kontrolu nad přístupem uživatelů k internetu. Zakázat všem uživatelům na učebně přístup k internetu nebo jen jednotlivcům, vytvářet seznam webových stránek nebo klíčových slov, který bude sloužit k zablokování a evidenci návštěvnosti těchto webových stránek. Z evidence návštěvnosti webových stránek jednotlivých uživatelů, by měl kantor mít přehled o stránkách, které uživatelé navštěvují.

V dnešní době se mezi žáky rozmáhá trend, poškozovat majetek, obcházet pokyny kantorů, úmyslně pozměňovat systémová nastavení, instalovat nežádoucí software. Kantor nemůže ve výuce ztrácet čas hlídáním uživatelů, neustálým zakazováním nebo složitým nastavováním, proto správa internetového provozu musí být pro kantora jednoduchá, přehledná a snadno dostupná. Při správě internetového provozu je nutné dbát na bezpečnost, ochranu před útoky z internetu, viry, ale také před útoky samotných uživatelů.

S nedostatkem internetové konektivity se potýká většina škol. Většina škol zapojených do projektu „ Internet do škol“, stále využívá připojení k internetu prostřednictvím ADSL modemu, které bylo součástí projektu. Důvodem je nedostupnost služeb ISP poskytovatelů. Jelikož projekt byl ukončen v roce 2005, pro dnešní požadavky je připojení nevyhovující. Klíčové bude vyřešit problém s nedostatkem internetové konektivity, neboť připojení k internetu prostřednictvím ADSL modemu je nedostatečné, pro provoz na počítačové učebně, natož pak pro celou organizaci.

3 Přehled aplikací

V přehledu se seznámíme s produkty nejen pro správu internetového provozu. Správa internetového provozu je ve většině programů doplněna o monitoring uživatelů a správu počítačové učebny. Správu počítačové učebny zahrbuje vypínání a zapínaní počítačových stanic, zobrazení aktuální plochy uživatele, zamezení ovládání počítače, zobrazení konkrétního obsahu na monitorech uživatelů, logování aktivity uživatele. Popis jednotlivých aplikací bude zaměřen na část řízení internetového provozu.

3.1 PC Control

PC Control 2 [13] je komerční aplikace dostupná pouze pro platformu Windows. Jedná se o komplexní řešení správy počítačové učebny. Z hlediska řízení internetu aplikace nabízí blokování celé učebny, blokování a monitorování přístupu k internetu jednotlivých uživatelů, blokování definovaných webových stránek. Aplikace se skládá s klientské a učitelské (serverové) aplikace. Aplikace komunikuje mezi sebou pomocí definovaných portů, které lze definovat při samotné instalaci. Aplikace pro svůj běh nevyžaduje administrátorská oprávnění.

3.1.1 Výhody

Komplexní řešení správy počítačové učebny a internetového provozu, podpora multimediální výuky a intuitivní ovládání.

3.1.2 Nevýhody

Pro každou počítačovou učebnu je nutné zakoupit zvlášť licenci a provést instalaci softwaru. Pro správný chod aplikace musí být aplikace spuštěna na učitelském počítači po celou dobu, jinak se neeviduje seznam navštívených stránek, nelze blokovat internet, ani blokovat webové stránky. Problémy s komunikací mezi stanicemi a učitelským počítačem. Vysoké nároky na šířku pásma počítačové sítě. Použitelnost aplikace do 25 stanic v jedné učebně.

3.2 Správce učebny

Správce učebny [14] je aplikace určena pouze pro platformu Windows. Je rozdělena na klientskou a učitelskou část. Program poskytuje základní funkce správy počítačové učebny. Od zapnutí nebo vypnutí počítačů, blokování vstupu ze strany uživatele, blokování internetu, zamezení přístupu k sociálním sítím, monitorování práce uživatelů. Konfigurace počítačové učebny se provádí z učitelského počítače zadáním seznamu názvů počítačů nebo načtením počítačů z Active Directory. Při instalaci se vytváří instalační balíček služby Microsoft Windows Installer. Řízení přístupu do aplikace je pomocí hesla účtu správce nebo učitele. Správce učebny je možné volitelně rozšířit o auditní a monitorovací funkce. Efektivní přínos má zejména v oblasti monitoringu a omezení používání USB paměťových zařízení, práci s nimi a také SW a HW auditu.

3.2.1 Výhody

Snadné nasazení klientské aplikace na stanice školní sítě pomocí vygenerovaného instalačního balíčku, Acitve Directory a Group Policy. Prostředí aplikace je uživatelsky přívětivé a intuitivní.

3.2.2 Nevýhody

Nefunkční blokace internetu bez spuštění aplikace na učitelském počítači. Časté výpadky připojení stanic k učitelské stanici. Vysoké nároky na šířku pásma počítačové sítě.

3.3 LanSchool

LanSchool [15] je komerční aplikace podporující platformy Windows, Linux, Mac OS, iOS, Android a Chomebook. Aplikace řeší komplexní správu počítačové učebny. Aplikace umožňuje blokovat internet, povolit práci jen na určitých webových stránkách a povolit spouštět jen povolené aplikace operačního systému. Učitel může využívat aplikaci Teacher's Assistant na zařízení iPad, která se spáruje s konzoli, která běží na platformě Windows nebo Mac OS. Řešení LanSchool nabízí Cloud Classroom, díky které není nutné aplikaci instalovat. Aplikace pracuje s podsítěmi a VLAN-y, pomoci technologií Multicasting a Direct Broadcasting. Využívá k tomu port 796.

3.3.1 Výhody

Nízké nároky na hardware klientské stanice, na šířku pásma (5-20% z používané šířky pásma konkurence). Komplexnost řešení správy počítačové učebny, podpora mnoha platforem a možnost Cloudového řešení.

3.3.2 Nevýhody

Chybějící česká lokalizace, složitější konfigurace aplikace.

3.4 Kerio Control

Kerio Control [16] (dříve označovaný WinRoute Firewall) je určený pro unifikované zabezpečení sítě. Jedná se o komerční aplikaci určenou do segmentu malé a střední organizace. Mezi základní funkce patří síťový firewall, směrovač, detekce a prevence útoků IPS, VPN server, rozložení zátěže internetového připojení s cílem maximalizovat šířku pásma a automaticky zálohovat internetové připojení. Do volitelných funkcí patří filtrování webových stránek a obsahu, integrovaný Sophos Gateway Antivirus, který kontroluje veškerou webovou komunikaci, přenosy FTP, e-maily, přílohy a stahované soubory. Veškerá správa a konfigurace se provádí z webového administračního rozhraní. Správu lze provádět kdykoli a odkudkoli, ze stolního počítače či z tabletu. Možností nasazení jsou Software Appliance, instalace jako samostatný operační systém nebo jako virtuální

zařízení Virtual Appliance určené k provozování na produktech WMware nebo Hyper-V. Poslední možností je hardwarové zařízení s optimalizovaným výkonem.

3.4.1 Výhody

Integrace uživatelů s Active Directory, nasazení virtuálních řešení, čímž odpadají náklady na pořízení nového hardwaru a spotřebu elektrické energie. Připojení z nezabezpečené sítě do organizace přes zabezpečenou síť VPN, integrace antivirové ochrany, filtrování webových stránek, obsahu a přehledné reportovaní uživatelských aktivit.

3.4.2 Nevýhody

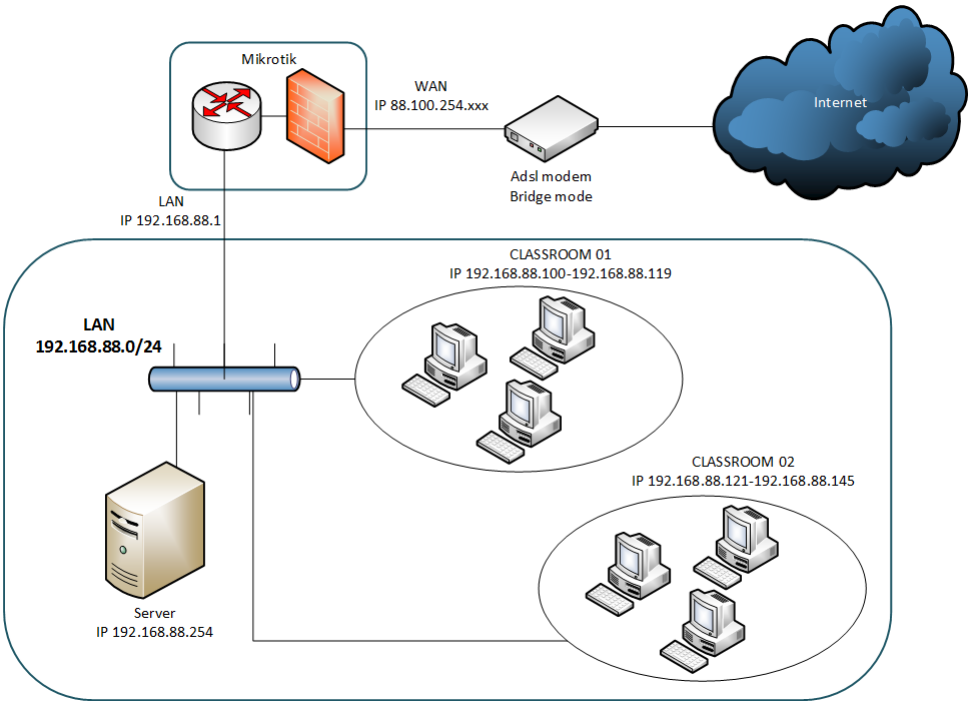
Vysoké pořizovací náklady a uzavřenost systému. Neumožňuje uživateli jednoduše blokovat internet, pro běžného uživatele příliš složité ovládání.

4 Architektura sítě

Při návrhu aplikace jsem bral zřetel na nejobecnější řešení, tak aby byl využitelný na libovolné síťové architektuře. Většina škol byla zapojena do projektu „Internet do škol“, proto každá škola byla připojena k internetu přes ADSL modem a jako směrovač se využíval server s operačním systémem Microsoft Windows Server 2003 Standard.

I když projekt skončil, tak tato skutečnost nadále ve většině případů přetrvává. Ze zastaralé technologie modemu a opeačního systému serveru, byl pro lepší správu, bezpečnost a monitoring vložen za ADSL modem aktivní prvek Mikrotik s operačním systémem RouterOS. Dále se server s operačním systémem Microsoft Windows Server 2003 vyměnli za nový nebo se upgradoval operační systém.

Celý návrh je tedy vyvíjen pro nejběžnější architekturu, která zahrnuje ADSL modem, Mikrotik, přepínač, Server a dvě počítačové učebny.



Obrázek 1: Ukázka architektury sítě

Jak již bylo zmíněno připojení k internetu je realizováno ADSL modemem. Modem je v režimu bridgw. Ten předává veškerá nastavení Mikrotiku na rozhraní PPPoE. Na rozhraní PPPoE je od poskytovatele internetu přidělena IP adresa. Mikrotik je ve stávající architektuře využíván jako směrovač a firewall. Ve vnitřní síti automaticky přiděluje IP adresy Server, pomocí protokolu DHCP. Dále má Server funkci DNS serveru pro překlad názvů počítačů na IP adresy.

5 Mikrotik - RouterOS

Systém je založen na Linuxu, ale je však zcela komerční. Ke komunikaci s operačním systémem RouterOS lze využít aplikace Winbox, Webfig, základní webové rozhraní, SSH, Telnetu a rozhraní Mikrotik API. Systém je koncipován na platformy i386, mips a powerpc. Distribuován je ve formě balíčků NPK, přeinstalovaného sytému RouterBoard nebo obrazu ISO. Mezi základní části systému patří firewall, routování, řízení šířky pásma, proxy server, virtuální privátní sítě, řešení bezdrátových sítí, skriptovací funkce, kompletní Hotspotové řešení, rozsáhlé možnosti logování, monitorování provozu. Zde výčet zcela nekončí. Systém je otevřený a je zde možnost využít přídavných aplikací. RouterOS lze provozovat na klasickém počítači kompaktibilní s architekturou i386, tak na speciálních platformách jako jsou routerbordy. [4] Informace o jednotlivých částech RouterOS byly čerpány z oficiálních stránek manuálu výrobce Mikrotik. [17]

5.1 Licence

Rozdělení licenci:

- Level 0 - licence zdarma, umožněna veškerá konfigurace, omezeno na 24 hodin
- Level 1 - demo licence, silně omezená, bez konfigurace bezdrátových rozhraní
- Level 2 - neexistuje
- Level 3 - určeno pro klientská zařízení, nepodporuje vytváření AP, podporuje směrovací protokoly
- Level 4 - nejběžnější a nejpoužívanější licence. Umožňuje veškerou konfiguraci zařízení, omezení jsou pro většinu uživatelů zanedbatelná
- Level 5 - využití ve zvlášť náročných případech, kde jsou nároky na vysoký počet uživatelů a VPN sítí
- Level 6 - bez jakýchkoliv omezení

5.2 API

U RouterOS lze využít přistup pomoci API rozhraní. Rozhraní umožňuje uživatelům vytvářet vlastní softwarová řešení pro komunikaci s RouterOS, shromažď'ovat informace, upravovat konfiguraci a správu routeru. API je dostupné pro většinu programovacích jazyku. API využívá port 8728. Komunikace je nešifrovaná a tím pádem je zde možnost odposlechu citlivých údajů. Proto je vhodné zabezpečit spojeni externími prostředky např. šifrované VPN spojeni, pravidly ve firewallu apod.. [3]

Komunikace se směrovačem se provádí zasláním věty ke směrovači a přijímání jedné nebo více vět na zpět. Věta je posloupnost slov ukončená slovem s nulovou délkou. Slovo je součástí věty složené z kódované délky získané z tabulky 1 a samotných dat. Schéma umožňuje kódování délky až 0x7FFFFFFFFF, i když je podporována pouze délka čtyř

Délka slova	Počet bajtů	Kódování
0 <= délka <= 0x7F	1	délka, nejnižší bajt
0x80 <= délka <= 0x3FFF	2	délka | 0x8000, dva menší bajty
0x4000 <= délka <= 0x1FFFFF	3	délka | 0xC00000, tři menší bajty
0x200000 <= délka <= 0xFFFFFFF	4	délka | 0xE0000000
délka >= 0x10000000	5	0xF0 a délka jako čtyři bajty

Tabulka 1: Kódování slov Mikrotik API

bajtů. Pokud první bajt slova je > = 0xF8, pak je vyhrazen kontrolní bajt. Po obdržení neznámého kontrolní bajtu, API klient nemůže pokračovat, protože neví, jak interpretovat následující bajty.
První slovo ve větě je vyhrazené samotnému příkazu, před kterým musí být znak „/“, pak následují argumenty příkazu. Příkaz má striktně dané pořadí:

- kódovanou délku
- prefix „/“
- upravené příkazy pro CLI

```
/user/active/ listen
/system/reboot
/ip/ firewall / filter /enable
```

Výpis 1: Ukázka kódu příkazu API

Struktura argumentu se skládá z pěti částí a mají striktně dané pořadí. Začíná kódovanou délkou slova, následuje počáteční znak „=“, následovaný názvem argumentu. Hodnotu argumentu od názvu argumentu, odděluje separační znak „=“. Argument nemusí mít žádnou hodnotu.

```
=address=10.0.0.1
=disable-running-check=yes
```

Výpis 2: Příklady argumentů bez kódované délky

Návratové slovo je odesláno pouze směrovačem a to na základě přijaté věty od klienta. První slovo odpovědi začíná „!“. Každá přijatá věta, generuje alespoň jednu odpověď (pokud není připojení přerušeno). Poslední odpověď každé věty začíná slovem „!done“. Chyby a výjimky začínají slovem „!trap“. Odpověd obsahující data začíná „!re“. Je-li ukončené spojení s Mikrotik API, RouterOS posílá odpověď „!fatal“.

```
<<< /ip/address/add
<<< =address=192.168.88.1
<<< =interface=asdf
<<<
```

```
>>> !trap
>>> =category=1
>>> =message=input does not match any value of interface
```

Výpis 3: Komunikace protokolu Mikrotik API

5.3 Firewall

5.3.1 Address list

Umožňuje uživateli vytvářet seznamy IP adres seskupených pod společným názvem.Tyto seznamy adres se pak dají využít, jako seznam zdrojových nebo cílových IP adres pro Firewall Filtr, Mangle a NAT. Seznam může obsahovat statické nebo dynamické záznamy IP adres.

5.3.2 Connections

Umožňuje sledovat a filtrovat navázané aktivní spojení.

5.3.3 Filter

Firewall provádí filtrování paketů čímž nabízí bezpečnostní funkce, které se používají k řízení toku dat do, z a přes router. Spolu s NAT, zamezuje neoprávněnému přístupu k přímo připojeným sítím, k routeru samotnému a dále slouží jako filtr odchozího provozu. Úkolem Firewallu je držet citlivá data uvnitř sítě od hrozeb z vnějšku. Firewally jsou používány jako prostředek k zabránění nebo minimalizaci bezpečnostních rizik spojených s připojením do jiných sítí. Správně nakonfigurovaný firewall, hraje klíčovou roli, v nasazení efektivních a bezpečných síťových infrastruktur. Firewall pracuje pomocí pravidel. Pravidla pro filtrování jsou seskupeny v řetězcích. Paket musí být porovnán s jedním společným kritériem v jednom řetězci, pak přejde k dalšímu zpracování některých jiných společných kritérií do jiného řetězce.

K dispozici jsou tři předdefinované řetězce, které nemohou být odstraněny:

- Input - slouží k zpracování paketů, které vstupují do routeru přes jedno z rozhraní s cílovou IP adresou, která je jedním z adresy routeru. Pakety procházející přes router, nejsou zpracovány v rozporu s pravidly vstupního řetězce.
- Forward - slouží k zpracování paketů procházejících routerem.
- Output - slouží ke zpracování odchozích paketů routeru na výstupu prostřednictvím jednoho z rozhraní. Pakety procházející přes router, nejsou zpracovány v rozporu s pravidly výstupního řetězce.

Při zpracování řetězce jsou pravidla převzaty z řetězce v pořadí, v jakém jsou uvedeny, tedy z shora dolů. Pokud paket odpovídá kritériím pravidla, pak se na něm provede zadaná akce a žádná další pravidla nejsou zpracovávány v tomto řetězci (výjimkou je akce passthrough). Pokud paket neodpovídá žádnému pravidlu uvnitř řetězce, pak je přijat.

5.3.4 Layer 7

Layer 7 shromažďuje prvních 10 paketů spojení nebo první 2KB spojení a hledá vzorek v získaných datech. Pokud vzorek není nalezen v získaných datech, zastaví kontrolu dalšího. Přidělená paměť se uvolní a protokol je považován za neznámý. Je třeba vzít v úvahu, že mnoho spojení významně zvyšují využití paměti a procesoru. Aby k tomu nedocházelo, je třeba snížit množství dat předávaných do Layer 7.

Další podmínkou je, že Layer 7 musí vidět oba směry provozu (příchozí i odchozí). Ke splnění tohoto pravidla Layer 7, by měl být požadavek nastaven v řetězci Forward.

5.3.5 Mangle

Jedná se o značkovač, který označuje pakety pro další zpracování pomocí speciálních značek. Těchto značek využívá např. Queue trees, NAT, routování. Značky existují pouze v routeru, nejsou přenášeny do sítě. Navíc, se používá k úpravě některých polí v hlavičce IP, jako jsou TOS (DSCP) a TTL pole.

5.4 NAT

Je internetový standard, který umožňuje počítačům na lokálních sítích používat jednu sadu IP adres pro interní komunikaci a další sadu IP adres pro externí komunikaci.

Existují dva typy NAT:

- Source NAT nebo srcnat - překlad zdrojových adres - pakety iniciované privátní sítí jsou bránou pozměněny za veřejnou adresu, která je směrovatelná v rámci celého internetu.
- Destination NAT nebo dstnat - překlad cílových adres - změna paketů určených pro cílovou privátní síť. Pomocí dstnatu je možné zpřístupnit počítač z privátní sítě do celého internetu.

Mezi zvláštní případy cílového NATu můžeme považovat funkce redirect a masquerade.

- Redirect v hlavičce paketu pozměňuje pouze cílový port, cílová adresa zůstává stejná.
- Masquerade při průchodu paketu se nastaví v hlavičce adresa daného rozhraní, jímž paket právě prochází. RouterOS eviduje v databázi seznam těchto překladů, aby mohl přicházející pakety převést do korektního stavu.

5.5 Routing

RouterOS nabízí dva základní typy routování paketů. A to dynamické a statické. Statickými routami rozumíme běžné routy přidané správcem k směrování provozu do vzdálených sítí a pro účel konfigurace výchozích bran v síti. Dynamické routování je automatické přidání routy po přiřazení IP adresy k fyzickému adaptéru. Je zde možnost využít

speciálních protokolů pro dynamické směrování jako jsou například BFD, BGP, IGMP-Proxy, MME, Multicast, OSPF, Prefix list, RIP. Nechybí podpora IPv6 dynamických routovacích protokolů.

5.6 VPN

Virtuální privátní sítě používají veřejné sítě k simulaci vnitřních nebo soukromých sítí. Umožňují tedy uživatelům privátní komunikace užívat veřejnou nebo sdílenou síťovou infrastrukturu. Privátní znamená bezpečné oddělení provozu od ostatních uživatelů. Rozvoj těchto sítí byl závislý na rozšíření protokolu IPSec. Softwaroví klienti jsou nyní dostupní na všech OS. VPN negarantuje šířku pásma a dobu odezvy. Použití je tam, kde není zapotřebí stálého spojení. Aby provoz zůstal privátním, musí být provoz šifrován. VPN tedy řeší problém přímého přístupu přes Internet. Každá VPN musí obsahovat tři bezpečnostní prvky:

- Zapouzdření IP (encapsulation) - IP paket odesílaný z jedné části sítě LAN se zabalí do jiného paketu. Ten pak putuje do cíle, kde se tento paket rozbalí a vznikne původní paket.
- Šifrována autentizace
- Šifrování dat

RouterOS podporuje tunely bod bod (OpenVPN, PPTP, PPPoE, L2TP, SSTP), jednoduché tunely (IPIP, EoIP) a Ipsec.

5.7 DHCP server

Jeho úkolem je automatická konfigurace počítačů připojených do sítě. Komunikace probíhá ze strany klienta na UDP portu 68 a server naslouchá požadavkům na UDP portu 67. DHCP server přiděluje počítačům IP adresy, masku sítě, vychozí bránu a adresy DNS serverů. Platnost přidělených údajů je omezená, proto je na počítači spuštěn DHCP klient, který jejich platnost prodlužuje. Významným způsobem tak zjednodušuje a centralizuje správu počítačové sítě (například při přidávání nových stanic, hromadné změně parametrů, skrytí technických detailů před uživateli). RouterOS dále nabízí DHCP client, relay, statické a dynymické zapůjčky, volitelné nastavení, rezervace, podporu IPv6 a Hotspotů.

5.8 Queues

Řízení datového toku v RouterOS řeší následující činnost:

- Omezování (popř. upřednostňování) rychlostí jednotlivých IP.
- Omezování rychlosti (popř. upřednostňování) jednotlivých protokolů, portů a na nich běžících služeb.
- Vytváření sdílených linek.

- Řízení provozu P2P systémů.
- Vytvářet statistiky o přenesených datech.

5.8.1 Mechanismy omezování

RouterOS podporuje několik mechanismů omezování a řízení síťového provozu. Např.

- PFIFO a BFIFO (packets first-in first-out) a bfifo (bytes first-in first-out)
- SFQ (stochastic fair queueing)
- RED (random early detection)
- PCQ (per connection queue
- HTB (hierarchical token bucket)

5.8.2 Rozdíl mezi Simple Queues a Queue Tree a případy jejich použití

Simple Queues jsou již dle názvu určeny pro základní řízení provozu. Pokud chcete pouze omezovat rychlosti jednotlivým IP adresám, popř. jejich rozsahům (pouze v rámci subnetů sítě), je ideální použít Simple Queues.

Queue Tree slouží pro sofistikované řízení toků. Pokud chcete vytvářet sdílené linky, upřednostňovat jednotlivé protokoly nebo služby běžící na určitých portech, budete muset použít Queue Tree. Jejich nevýhoda je v určité složitosti konfigurace.

Simple Queue a Queue Tree mohou být zkombinovány pro vzájemný provoz vedle sebe. Je však nutné mít na paměti, že vzájemnou kooperaci je vhodné vyzkoušet.

5.9 Webproxy

Web Proxy server funguje jako prostředník mezi klientem a cílovým počítačem. Proxy sleduje požadavky přicházející od klienta přes protokoly HTTP, FTP a ukládá si kopie odpovědí do mezipaměti. Pokud přijde jiný požadavek na stejnou adresu URL, použije uloženou kopii, místo toho, aby znovu žádal původní server.

- Regular proxy - uživatel si sám určí, zda bude využívat proxy server.
- Transparent proxy - bez nutnosti konfigurace ve webovém prohlížeči uživatele. Transparentní proxy server nemění požadované URL nebo odpovědi. RouterOS vezme všechny HTTP požadavky a přesměruje je na lokální proxy službu. Tento proces je pro uživatele zcela transparentní (uživatelé neví nic o proxy serveru, který je umístěn mezi nimi a původním serverem).

5.9.1 Další funkce:

- podpora ukládání mezipaměti na externí uložiště
- získávání a uchovávání informací o provozu proxy serveru
- podpora nadřazeného proxy serveru
- podpora protokolu SOCKS a DNS statických záznamů

5.9.2 Možné způsoby využití proxy serveru:

- urychlení přístupu ke zdrojům
- filtrování webového obsahu (podle URL nebo požadavků HTTP)
- skenování odchozího obsahu
- řízení přístupu na nevhodné webové stránky
- omezení síťových služeb či obsahu
- klienti se díky Web Proxy stávají pro cílové servery anonymní

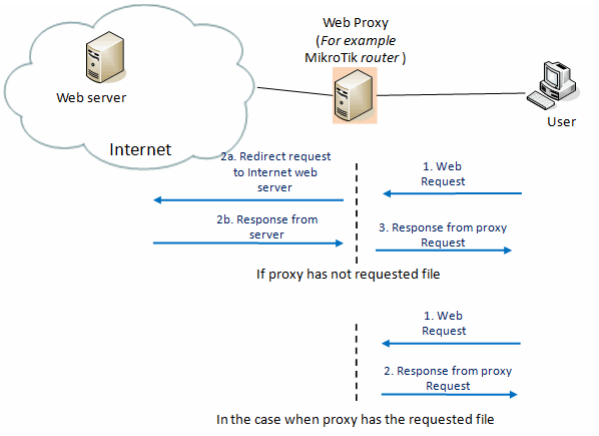


Figure 10.1. Web proxy basic operation scheme

Obrázek 2: Základní schéma provozu Webproxy

5.9.3 Specifické příkazy

- Access list – je realizován stejným způsobem jako pravidla firewallu, zpracované z vrcholu až na dno. První odpovídající pravidlo určuje, co se má udělat s tímto připojením. Shodu můžeme najít ve zdrojové adrese, cílové adrese, zdrojovém portu, podřetězci požadované adresy URL nebo metodou požadavku. Pokud se spojení

shoduje s pravidlem, vlastnost akce tohoto pravidla určuje, zda bude spojení povoleno nebo odepřeno. Pokud připojení neodpovídá žádnému pravidlu, bude povoleno.

- Direct Access - pokud je nastavena parent-proxy, může se proxy server pokusit předat žádost na nadřazený proxy server nebo se pokusit vyřešit připojení k požadovanému serveru přímo. Direct Access je řízen stejně jako Access list, s výjimkou akčního argumentu. Na rozdíl od Access listu, je výchozí akce odepřít. To nastává, pokud nejsou zadány žádné pravidla nebo konkrétní požadavek neodpovídá žádnému pravidlu.
- Cache access list - specifikuje, které požadavky (domény, servery, stránky) budou ukládány do mezipaměti lokální proxy, a které ne. Tento seznam je implementován přesně stejným způsobem jako Access list. Implicitní je nastaveno ukládání objektů do mezipaměti.

5.10 Tools

RouterOS obsahuje nesčetné množství funkcí. Vypsat však tento seznam není předmětem zájmu. Zde se zaměříme pouze na několik užitečných a často používaných nástrojů, doplněných o stručný komentář.

- Ping – tato utilita nám umožní ověřit přímo s RouterOS dostupnost zařízení pomocí ICMP protokolu.
- Bandwidth test – test propustnosti linky mezi dvěma zařízeními Mikrotik.
- Torch – pomůcka pro sledování provozu na routeru. Pomocí filtrů zpřesňujeme podmínky sledování provozu na určitém rozhraní.
- Telnet, ssh – obsahuje klientskou i serverovou část. Slouží k přístupu a konfiguraci RouterOS.
- E-mail a SMS odesílání – pomocí této služby je možno odesílat reporty, výpisy logů na email nebo formou SMS. Emailové odesílání se často používá k zálohování RouterOS.
- NTP client/server – tato služba umožňuje využít RouterOS, jako zdroje času. Slouží k synchronizaci času počítačů v síti.

6 Analýza a návrh vlastního řešení

Cílem analýzy je vytvoření seznamu požadavků, které slouží jako podklad, ke zpracování návrhu rozhraní pro správu routeru na platformě Mikrotik.

6.1 Analýza požadavků

6.1.1 Společné požadavky

Využití stávající síťové architektury popsané v kapitole Aplikace bude komunikovat s routrem pomocí rozhraní Mikrotik API. Měla by být nezávislá na operačním systému a využít prostředků operačního systému. Přístup do aplikace by měl být autorizován uživatelským jménem a heslem. Uživatelé budou mít přiřazené role, které definují omezení v používání aplikace.

6.1.2 Požadavky uživatelského rozhraní

Uživateli umožnit blokovat internetový provoz, pro konkrétní IP adresy nebo pro definovaný rozsah adres. Blokování internetu rozšířit o možnost nastavit čas blokování. Zamezit přístup ke konkrétním webovým stránkám nebo blokovat webové stránky na základě seznamu klíčových slov. Spravovat seznam blokovaných webových stránek. Zobrazit statistiku blokovaných a navštívených webových stránek.

6.1.3 Požadavky konfiguračního rozhraní

Konfigurační rozhraní má rozšířit uživatelské rozhraní o nastavení aplikace, správu uživatelů a záznam jejich práce s aplikací.

6.2 Analýza stávajícího řešení

Ve stávající počítačové síti popsané v kapitole 4 je hlavní prvkem směrovač na platformě Mikrotik. Router slouží čistě jako brána do internetu a firewall. Spolu se směrovačem nalezneme v sítí Server s operačním systémem Windows Server Standard 2008 R2 nebo Windows Server Standard 2012. Tento server se stará, o přiřazování IP adres, počítačům pomocí služby DHCP server. Server slouží dále jako řadič domény, DNS server a File server. Ze stávajícího řešení je patrno, nevyužití potenciálů jednotlivých zařízení.

6.3 Sledování internetového provozu

Díky výpisu logu Webproxy, získáme seznam navštívených stránek. Ale Webproxy nám nezajistí výpisy navštívených šifrovaných stránek. Řešením sledování šifrovaných stránek by bylo, nasazení tzv. Man in the middle proxy. Proxy se snaží odposlouchávat komunikaci mezi účastníky tím, že se stane prostředníkem komunikace. Odchytí veřejný klíč jednoho účastníka, nahradí ho svým podvrženým veřejným klíčem. Stejně podvrhne veřejný klíč druhého účastníka. Oba účastnící se domnívají, že mají veřejné klíče toho

druhého. Cokoliv si účastníci pošlou, proxy dešifruje, přečte, znovu zašifruje a pošle druhému. Dokonce může komunikaci pozměnit nebo přesměrovat, jako by byla pravá. Oba účastníci nic nepoznají. [22]

Ale zde narážíme na legislativu ochrany osobních údajů. „*Sledovat používání webových stránek zaměstnanci pro účely zaměstnavatele tedy možné není, pokud nejsou splněny zákonem stanovené podmínky, tj. závažný důvod spočívající ve zvláštní povaze činnosti zaměstnavatele. Pod tím si lze představit například mezinárodní bankovní převody nebo dozor nad prací vězňů.*“ [20]

Pro účely sledování navštívených stránek se spokojíme s výpisem logu Webproxy. Výpis se bude filtrovat pomocí regulárního výrazu pro ukládání URL adres.

```
^http \:\/\/[ a-zA-Z0-9\-\.]+\.[a-zA-Z]{2,3}\/$
```

Výpis 4: Ukázka regulárního výrazu

6.4 Návrh vlastního řešení

Důraz návrhu řešení byl kladen na využití stávající síťové architektury, dostupných hardwarových a softwarových zdrojů. Instalace dodatečných programů by se měla řídit licencí freeware.

Pro potřeby nezávislosti aplikace na operačním systému a modulárnosti systému bude rozhraní nasazeno jako webová aplikace. K implementaci bude použita technologie ASP.NET s využitím návrhového vzoru Model-View-Controller, dále frameworku ASP.NET MVC 4 a programovacího jazyka C#. Díky návrhovému vzoru MVC je zajištěno oddělení uživatelského rozhraní od řídící logiky a datového modelu. Čehož lze využít k dalšímu rozšíření aplikace.

Pro komunikaci s operačním systémem RouterOS bude využito rozhraní Mikrotik API. Pro správnou funkci webová aplikace, se budou skrze Mikrotik API používat tyto RouterOS moduly Webproxy, Firewall Filter, NAT, Log a Users. K zvýšení bezpečnosti komunikace webové aplikace s RouterOS, je nutné zařadit mezi Mikrotik a webovou aplikaci tzv. TCP proxy. Webová aplikace nebude komunikovat s Mikrotikem přímo, ale prostřednictvím TCP proxy, ta předá komunikaci Mikrotiku a naopak. Informací o navštěvovaných stránkách nám umožní služba, Syslog, která bude zapisovat výpisy logu Web Proxy do databáze. Pro správu přístupu do webové aplikace, se provede autentizace uživatele a následně přiřazení role. Proto je zapotřebí evidovat seznam uživatelů v databázi.

Aby byly využity stávající hardwarové a softwarové zdroje, nabízí se pro účely nasazení webové aplikace, využit dostupných služeb operačního systému Microsoft Windows Server. Jako webový server využijeme službu IIS (Internet Information Services). Pro ukládání logů z Webproxy a ze samotné aplikace, využijeme databázového serveru Microsoft SQL Server. Dále do databáze budeme potřebovat ukládat evidenci uživatelů a učeben. DHCP server bude využit k automatickému přidělování IP adres. Přidělené IP adresy je nutné zařadit do rezervace tak, aby jednotlivým počítačům byla přidělena pokaždé stejná IP adresa.

7 Vlastní řešení

7.1 Požadavky na konfiguraci

K běhu aplikace je zapotřebí dodatečných konfigurací, instalací aplikací a zprovoznění služeb.

7.1.1 Konfigurace Mikrotik - RouterOS

Konfigurace RouterOS je směřovaná na správce sítě, který zajistí požadovanou konfiguraci.

- *povolit rozhraní Mikrotik API* -v základním nastavení je rozhraní vypnuté. Pro větší bezpečnost, nastavíme přístup k rozhraní pouze z TCP proxy a změníme výchozí port 8728 na 18728. Tím zamezíme nežádoucímu odposlechu na standardním portu.
- *vytvořit novou skupinu a do ní přiřadit nového uživatele* - ke komunikaci s Mikrotik API vytvoříme skupinu WebAPP, která bude mít přístup k rozhraní API. Skupině nesmíme zapomenut přidat práva číst a zapisovat. Do skupiny přidáme nově vytvořeného uživatele WebUser. Komunikace, mezi webovou aplikací a Mikrotik API, bude prováděna přes uživatele WebUser.
- *konfigurace Webproxy a logů* - pro získávání informací o navštívených stránkách uživatelů. Nejprve musíme zapnout samotnou Webproxy. Výpisy se budou posílat na Syslog integrovaný ve službě ProxyService. Nastavení zahrnuje konfiguraci akce remote, zadáním IP adresy zařízení, na němž je služba ProxyService spuštěna a portem, na kterém Syslog naslouchá. Standardně je to port 514.
- *přidat pravidlo firewallu* - v sekci IP/Firewall/Filter Rules je nutné vytvořit pravidlo řetězce forward, k blokování internetového provozu, ze zdrojového adresního listu BLOCK. Pravidlu přiřadíme první pozici.

7.1.2 Server s operačním systémem Windows Server 2008 a vyšší

Požadavky na konfiguraci a instalaci softwaru jsou směřovány na systémového administrátora.

Pro provoz webové aplikace musí být na serveru instalovány role Internet Information Services, DHCP server, DNS server a platforma Microsoft .NET Framework 4.5. Dále bude zapotřebí instalovat Microsoft SQL Server 2008 Express a novější.

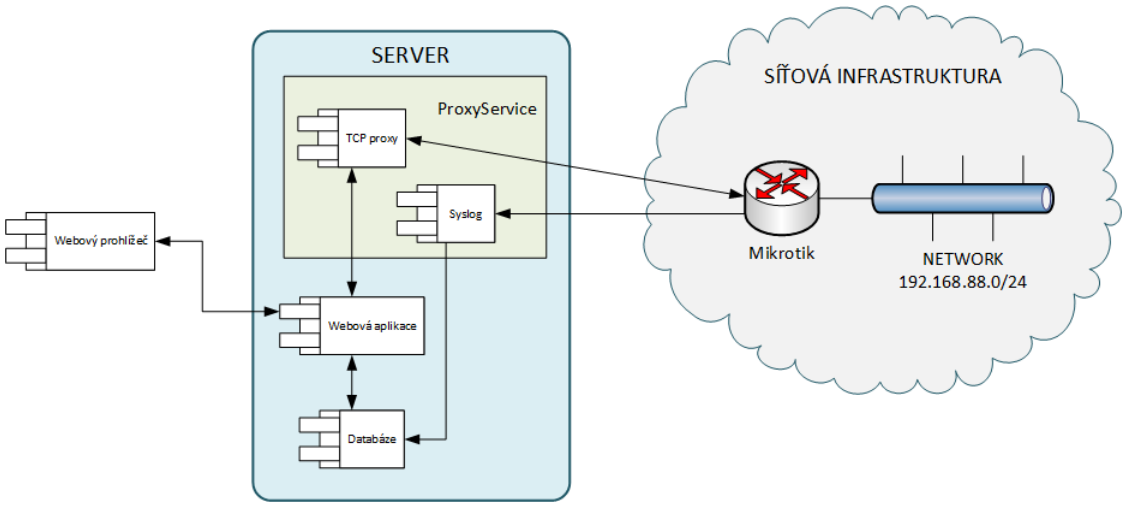
Databázovou strukturu vytvoříme pomocí skriptu. Skript vytvoří nejen tabulky PROXYLOG, LOGMK, CLASSROOMS, USERS, ale i výchozího uživatele webové aplikace „admin“, který bude mít roli Administrators.

DHCP server je třeba nakonfigurovat tak, aby jednotlivým učebnám přiřadil IP adresy počítačů v řadě za sebou. Pokud nebude využit server DHCP, musí být IP adresy zadány do počítačů ručně. V neposlední řadě zbývá povolení potřebných portů brány firewall serveru.

Pokud by nebyl k disposzici žádný server s operačním systémem Windows Server 2008 a vyšší, můžeme aplikaci nasadit na počítač s operační systémem Windows 7 Professional. Využije se integrovaný webový server IIS Express, naistaluje se Microsoft SQL Server 2008 Express a novější. DHCP a DNS server nahradíme službami Mikrotiku. Samozřejmě se musí naistalovat platforma Microsoft .NET Framework 4.5 a povolit potřebné porty na bráně firewall.

7.2 Architektura systému

Od začátku jsem se snažil aplikaci budovat, nejen pro obecnou síťovou architekturu popsanou v kapitole 4, ale pro různé síťové architektury, které obsahují samostatné servery, pro jednotlivé služby, jako jsou SQL Server, webový server, DHCP server. Proto jsem aplikaci, shluknul do jednoho bodu, aby byl systém centralizovaný, bezpečnější a nedrobil se.



Obrázek 3: Ukázka architektury systému

7.3 Mikrotik

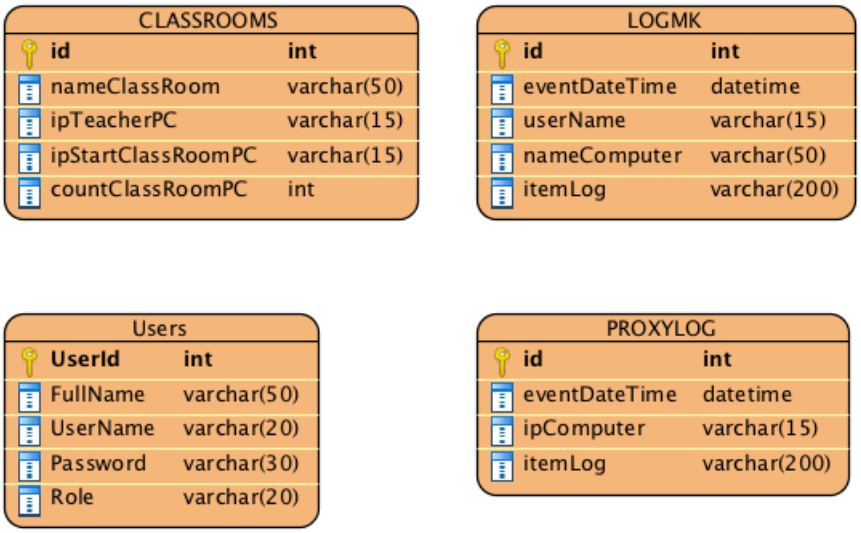
Srdcem celého sytému je prvek Mikrotik s operačním systémem RouterOS. Pro komunikaci s RouterOS se využívá Mikrotik API. Při implementaci rozhraní Mikrotik API jsem využil třídu MK, kterou výrobce na svých webových stránkách dává k dispozici, pro jednotlivé programovací jazyky. Třída MK pro programovací jazyk C# je dostupná z [18].

Pro správnou funkčnost třídy MK, musí být k dispozici jmenné prostory System.IO a System.Net.Sockets.

7.4 Databáze

K ukládání všech dat aplikace se stará databáze Logger. Databáze obsahuje tabulky CLASSROOM, LOGMK, PROXYLOG a Users. Tabulka CLASSROOM eviduje parametry nastavení počítačové učebny. Nejvíce záznamů se zapisuje do tabulky PROXYLOG, protože obsahuje logové výpisy z Webproxy. Tyto výpisy, do tabulky zapisuje Syslog, který je součástí služby ProxyService.

V tabulce LOGMK se evidují záznamy práce uživatelů s aplikací. Protože v záznamech tabulky LOGMK se budou evidovat, pouze IP adresy učitelských stanic, ve většině případů se jedná o dvě stanice, není třeba tabulky PORXYLOG a LOGMK propojovat.I když obě tabulky obsahují záznam IP adresy. Pro evidenci uživatelů aplikace slouží tabulka Users.



Obrázek 4: Ukázka databázové strukury

7.5 Služba ProxyService

7.5.1 Popis služby

Služba plní důležitou roli celého sytému. Pro různé síťové architektury služba ProxyService shlukuje celý systém, do jednoho uzlu tak, aby se systém nedrolil a byl bezpečnější. Služba má dvě části.

7.5.2 TCP proxy

Komunikace s Mikrotik API probíhá pomocí vět, proto není žádným způsobem chráněna před útoky nebo odposlechem. Pro posílení bezpečnosti komunikace mezi webovou aplikací a Mikrotik API, se mezi ně vloží TCP proxy. To zajistí, že Mikrotik API bude komunikovat s webovou aplikací pouze skrz TCP proxy. Dále se nastavíme, aby API bylo povolené pouze pro IP adresu TCP proxy.

7.5.3 Syslog

Syslog má za úkol, ukládát Webproxy logy z Mikrotiku do databáze. Naslouchá na portu 514 a čeká na data z Mikrotiku. Příchozí data z Mikrotiku nejprve přijme, po té je převede do srozumitelné podoby a následně uloží do databáze. Pokud služba zaznamená výjimku, zapíše ji do textového souboru, pro pozdější analýzu.

7.5.4 Implementace služby

Při implementaci TCP proxy serveru jsem vycházel z předlohy dostupné z [7]. Směrování paketů má na starost třída TcpForwarder.Je spuštěna v samostatném vlákně, se čtyřmi vstupními parametry. IP adresa, na které je služba spuštěna (nejčastěji localhost), port rozhraní Mikrotik API nastavené ve webové aplikaci, IP adresa Mikrotiku a cílový port Mikrotik API.

Syslog jsem implementoval dle předlohy [6]. Z původního kódu jsem odebral část, sloužící k odesílání emailů a zápisu logů do csv souboru. Pro potřeby bylo nutné doprogramovat ukládání logů do databáze. K ukládání do databáze byla využita třída database, kterou jsem převzal z předmětu DAIS a dbSyslog. Pro uložení záznamu do databáze slouží metoda handleLog(). Protože výpisy z logu Web Proxy jsou obsáhlé a při plném obsazení počítačových učeben, by mohla služba Syslog kolabovat, je metoda handleLog() a následně volaná metoda insert() třídy dbSyslog, spouštěná s využitím vláken. Ale nejprve je zapotřebí přijatý textový řetězec zpracovat do srozumitelné podoby. K tomu slouží parametrický konstruktor třídy Syslog. Třída Syslog reprezentuje data a třída database logiku.

V případě, že dojde k výjimce, bude prostřednictvím třídy errorLogWriter, výjimka zaznamenaná do textového souboru

7.6 Webová aplikace

Popis jednotlivých komponent webové aplikace.

7.6.1 Správa uživatelů

Správa uživatelů eviduje uživatele a jejich role. Přístup k jednotlivým částem aplikace se řídí, na základě ověřené autorizace a role uživatele. Přístup k jednotlivým částem aplikace je realizovaná, na úrovní uživatelského rozhraní a Accountcontrolleru. Z hlediska

bezpečnosti jsou veškerá hesla v aplikaci šifrovaná algoritmem md5. Uživatelé jsou uloženi v databázi v tabulce Users.

Správu uživatelů má na starost AccoutController. AccountController zpracovává požadavky na výpis uživatelů, přihlášení do systému, odhlášení ze systému, zakládá, upravuje a maže uživatelské účty

7.6.2 Implementace AccoutController

- Vypsání seznamu uživatelů se provede pomocí metody Index(). Metoda se připojí k databázi, načte si seznam uživatelů z tabulky Users a předá výsledek uživatelskému rozhraní (Account).

- V případě přihlášení je volaná metoda Login(). AccountControlleru je předán objekt typu LoginModel a řetězec s adresou návratu. Nejprve se musí pomocí metody encrypt() třídy EncryptDecrypt zakódovat heslo a následně se ověří, zda se uživatelské jméno a heslo v databázi vyskytuje.

- Odhlášení ze systému se provádí metodou LogOff(). Ta zavolá metodu SignOut(), dojde k odhlášení a přesměrování na úvodní obrazovku .

- Vytvoření uživatele se provádí zavoláním akční metody Regiter(). Metodě je předáván objekt typu RegisterModel. Aby bylo možné uložit objekt z třídy RegisterModel do databáze, musí se objekt přetypovat na objekt třídy Users. U přetypovaného objektu se zakóduje heslo a následně se objekt uloží do databáze.

- Při úpravě údajů uživatelského účtu, se AccountControlleru předává pouze id uživatelského účtu. Nejprve se z databáze načtou údaje o uživatelských účtech a porovnají se na základě identifikátorů. Vytvoří se objekt třídy Users, pak se přetypuje na objekt třídy RegisterModel a ten se předá uživatelskému rozhraní. Po provedení změn, se objet třídy RegisterModel přetypuje zpět na objekt třídy Users, zašifruje se heslo a uloží do databáze.

- Smazání se provede zavoláním akční metody Delete(), s předaným parametrem id. Id se ověří v databázi, pokud je záznam nalezen, pak je následně smazán.

7.7 Řízení internetového provozu

Tato komponenta systému je učena k řízení internetového provozu. Na tuto komponentu bude kladen důraz neboť je to funkcionalita přímo definovaná v zadání této práce.

K řízení internetového provozu se využívá BlockInternetController. Ten se stará o blokování internetu na celé učebně, jednotlivcům a blokování internetu s časovačem 15, 30 a 45 minut. A hlavně nám zobrazuje stav blokování internetu na učebně.

7.7.1 Implementace BlockInternetController

- O zobrazení stavu blokování internetu na učebně se stará akční metoda Index(). Nejprve si metoda zjistí z interní databáze parametry, pro připojení k rozhraní Mikrotik API. Tyto parametry předá konstruktoru objektu třídy MikrotikSettings. Uloží si IP adresu přihlášené stanice. Tuto IP adresu porovná s IP adresou učitelské stanice uložené v databázi. Pokud se shoduje, zavolá se metoda CheckBlock(), předají se ji objekty tříd MikrotikSettings a CLASSROOMS. Metoda CheckBlock() zavolá metodu GetIpRangeCollection() s parametry počáteční IP adresou a počtem počítačů v učebně. Metoda GetIpRangeCollection() vrátí metodě CheckBlock() kolekci IP adres počítačů v učebně. Nyní se vytvoří objekt třídy MK z předaného objektu metodou CheckBlock(). Pomocí metody Send() objektu třídy MK, se odešlou příkazy na Mikrotik. Metoda Read() objektu třídy MK přečte odpověď z Mikrotiku. Odpověď se porovná s vrácenou kolekcí IP adres. Pak se všechny IP adresy i ty bez shody, vloží do kolekce a ta se předá BlockInternetControlleru. BlockInternetControlleru předá kolekci uživatelskému rozhraní BlockInternet.

- Metody zablokovat nebo odblokovat internet jsou si velice podobné. Proto budu popisovat jen zablokování internetu. Metoda se jmenuje BlockAll() a má parametr ids, který je metodě předán z uživatelského rozhraní BlockInternet. Parametr identifikuje, o jakou učebnu se jedná. Pak se zavolá metoda BlockAll() třídy Commnads s parametrem nastavení Mikrotiku a objektem třídy CLASSROOMS. BlockAll() třídy Commnads následně volá metodu GetIpRange, ta vrací pole řetězců IP adres. Nyní se vytvoří instance třídy MK a metodou Send() se zapíše IP rozsah adres do seznamu BLOCK. U odblokování internetu se volá medoda UnBlockAll() a ta odebere rozsah IP adres ze seznamu BLOCK.

- Odblokovaní a zablokovaní jednotlivých IP adres je opět reverzní. Popíši odblokování. CurrentBlock() dostane z webového rozhraní parametr ids, sloužící k identifikaci IP adresy. Provede se inicializace Mikrotiku a zavolá se metoda UnBlockCurrent() s parametry objekt tříd MikrotikSettings a IP adresa.

7.8 Správa počítačových učeben

Opět se jedná o důležitou komponentu. Využívá se k identifikaci učitelského počítače. Při přidání nebo odebrání počítačové učebny se nejen pracuje s údaji v databázi, ale i s pravidly firewallu a NATu na Mikrotiku. Pravidla slouží k registraci učebny do systému blokování internetu.

7.8.1 Implementace ClassRoomsController

Vypsání seznamu učeben zahrnuje pouze načtení dat z tabulky CLASSROOMS a předání uživatelskému rozhraní, zavoláním akční metody Index(). Přidání nové učebny se realizuje akční metodou Create(). Je ji předán z uživatelského rozhraní objekt třídy CLASSROOMS. V tabulce CLASSROOMS se spočítají záznamy, součet se navýší o 1 a použije

se jako id nové učebny. Pak se učebna uloží do databáze a zavolají se metody RegisterClassRoom() a RegisterRedirect(). Předány budou parametry pro vytvoření objektu třídy MK a objekt třídy CLASSROOMS.

Metody RegisterClassRoom() a RegisterRedirect() pomocí metody Send() zapíši pravidla do Mikrotiku. Obdobným způsobem se provádí odebírání učebny.

Změna učebny se provádí metodou Edit(), té je předán parametr id z uživatelského rozhraní. Z tabulky CLASSROOMS se načtou data odpovídající id učebny. Vytvoří se objekt třídy CLASSROOMS a ten se předá uživatelskému rozhraní. Po provedení změn se objekt zapíše do databáze.

7.9 Výpis seznamu navštívených stránek

Tato komponenta slouží k zobrazení seznamu navštívených stránek. Jedná se o načtení dat z tabulky PROXLOG. Vzhledem k množství záznamů muselo být do InternetTrafficControlleru implementováno stránkování, řazení a filtrování.

7.9.1 Implementace InternetTrafficControlleru

Načtení seznamu se provádí akční metodou Index(). Jedná se o klasický způsob načtení dat z databáze. Jak již bylo zmíněno výše, byly implementovány funkce pro snadnější a přehlednější prohlížení výpisu. Jako optimální hodnota položek na stránku se osvědčila hodnota 20.

7.10 Výpis logu Webové aplikace

Tato komponenta slouží čistě pro kontrolu práce uživatelů s webovou aplikací. Zaznamenávají se zde veškeré úkony provedené v aplikaci. Tato komponenta je velice podobná InternetTrafficControlleru. Liší se pouze v datové struktuře tabulky.

7.10.1 Implementace LogControler

Implemntace je totožná s implementací InternetTrafficControlleru.

7.11 Správa seznamu blokovaných stránek

Tato komponenta je poslední, která souvisí s správou internetového provozu. Umožňuje blokovat webové stránky na základě klíčových slov nebo zadáním konkrétních URL adres. K blokování se využívá funkce Access Webproxy Mikrotiku. ProxyAccessController nám umožní zadávat, odebírat, povolovat nebo zakazovat blokování webových stránek.

7.11.1 Implementace ProxyAccessController

- Vypsání seznamu blokovaných stránek se realizuje akční metodou Index(). Ta zavolá metodu GetAll() s předaným objektem třídy MikrotikSettings. Metoda GetAll() zavolá metodu Send() s požadovanými příkazy. Výstupem bude kolekce ob-

jektu třídy ProxyAccessModels. Tuto vrácenou kolekci pak controller předá uživatelskému rozhraní.

- Přidání záznamu do seznamu se realizuje akční metodou Create(). Vstupní parametr je objekt třídy ProxyAccessModels. Dále se inicializuje objekt třídy MikrotikSettings, který se předá, spolu s parametry obsažené v objektu třídy ProxyAccessModels, metodě AddProxyAccess(). Ta vytvoří instanci objektu MK a pomocí metody Send() přidá položku do seznamu.
- Odebrání, zakázání či povolení je velice podobné přidání záznamu. Rozdíl je, ve formulací příkazu metody Send();

7.12 Správa nastavení systému

Správa nastavení slouží pouze k nastavení aplikace. Pro uložení dat je použita lokální databáze. Je to z důvodu oddělení konfigurace od databázového serveru. Slouží zcela pro interní použití. V nastavení se zadává účet pro přístup k Mikrotik API, nastavení připojení k SQL databázi, nakonec IP adresa a port TCP proxy .

7.12.1 Implementace SettingController

Pro přidávání, odebírání, úpravu a zobrazení se využívají standardní nastavení pro práci s databázi, jen u přidání a editace je přidáno šifrování hesel s využitím metody encrypt().

7.13 Uživatelské Rozhraní

Uživatelské rozhraní bylo využito z frameworku ASP.NET MVC 4, protože v základu bylo pro mé účely dostatečné a přehledné. Jednotlivé stránky jsem si podle potřeb upravil a pozměnil jsem kaskádové styly. Využitím uživatelských rolí, jsem rozdělil uživatelské rozhraní na tři části, dle jednotlivých rolí. Použití autentizace na úrovní uživatelského rozhraní, definujeme, které prvky rozhraní se nebudou zobrazovat. Což není až tak bezpečné. Autentizaci nad controllery, definujeme přístup k jednotlivým metodám. Kombinace obou způsobu autentizace umožňuje širokou škálu možností, jak dostatečně aplikaci zabezpečit.

Obrázek 5: Ukázka uživatelského rozhraní

7.13.1 Rozhraní role Administrators

Jak již nadpis napovídá, jedná se o rozhraní, kde uživatelé role Administrators mohou bez omezení používat všechny dostupné prostředky systému. Po přihlášení má uživatel v horní liště k dispozici celkovou nabídku. Ta obsahuje Home, Internet, Stránky, Proxy Log, Uživatelé, Učebny a Nastavení. Nabídka odpovídá použitým controllerům.

- Internet - má uživatel možnost blokovat internet celé učebně, jednotlivcům, nastavit časové blokování internetu. Samozřejmě získává přehled o stavu blokování internetu.
- Stránky – zde se uživateli zobrazí seznam blokovaných klíčových slov a webových stránek. Jsou seřazeny podle návštěvnosti sestupně. Má možnost přidávat, smazat nebo klíčové slovo či webovou stránku deaktivovat popřípadě znovu aktivovat.
- Proxy Log – ve výchozím zobrazení se uživateli zobrazí seznam navštívených stránek od nejnovějších po nejstarší. Kliknutím na popisy datum nebo stránky, lze měnit řazení seznamu. Podrobněji lze filtrování podle datumu od do.
- Uživatelé – zde je úplná podpora správy uživatelských účtů. Přidání, úprava a odebrání uživatelů.
- Učebny – slouží k správě počítačových učeben. Učebny lze přidávat, odebírat a upravovat.
- Log – eviduje aktivitu uživatelů webové aplikace, možnosti filtrvání a řazení jsou stejné jakou Proxy Logu.

- Nastavení – zde se nastavuje komunikace s TCP proxy, přístup k databázi a účet pro komunikaci s Mikrotikem.

7.13.2 Rozhraní role Power users

Práva a nabídku má uživatel role Power users podobnou jako role Administrators. Z menu nemá k dispozicipouze Nastavení. Je to z důvodu ochrany chodu aplikace. Dále nemá možnost mazat učebny a uživatele.

7.13.3 Rozhraní role Users

Uživatelé s rolí Users mají k dispozici pouze nabídky Home, Internet, Proxy Log, Stránky. Nemohou zadávat ani vypínat použití klíčových slov.

7.14 Obecná implementace

K implementaci webové aplikace jsem si vybral vývojové prostředí Visual Studio 2012 Profesional od společnosti Microsoft. Webové rozhraní využívá technologii ASP.NET. Modulárnost aplikace zajišťuje webový aplikační framework ASP.NET MVC 4, který implementuje návrhový vzor Model-View-Controller (MVC). K samotné implementaci jsem použil programovací jazyk C#. Pro uživatelské rozhraní jsem využíl syntaxe Razor.

8 Testovaní aplikace

Testování aplikace jsem provedl na architektuře popsané v kapitole 4.

8.1 Hardware

V této podkapitole si popíšeme hardwarové vybavení jednotlivých prvků v síťové architektuře.

8.1.1 Server

Model	HP ProLiant ML110 Generation 6
Název	SERVER01
Procesor	Intel Xeon X3430
Operační paměť	4GB
Pevné disky	2 x 500GB SATA, RAID 1
LAN	2 x 10/100/1000Mbps
IP adresa	192.168.88.254
Operační systém	Windows Server 2008 R2 Standard 64bit

Tabulka 2: Popis HW a SW Serveru

8.1.2 Mikrotik

Model	RB2011UAS
Procesor	Atheros 74K MIPS AR9344, 600MHz
Operační paměť	128MB
Pevné disky	128MB
LAN	5 x 10/100Mbps, 5 x 10/100/1000Mbps
IP adresa	192.168.88.1
Operační systém	RouterOS verze 6.16

Tabulka 3: Popis HW a SW Mikrotik

8.1.3 Stanice v učebně

Model	HP Pro 3500MT
Procesor	Intel Pentium G840, 2.80GHz
Operační paměť	4GB
Pevné disky	500GB, SATA3, 7200 ot./s.
LAN	1 x 10/100Mbps
IP adresa	192.168.88.99-119
Operační systém	Windows 7 Professional 32bit

Tabulka 4: Popis HW a SW stanic v učebně

8.1.4 Síťové prvky

Typ	přepínač
Model	Zyxel ES-1552
LAN	48 x 10/100Mbps, 2 x 10/100/1000Mbps, 2 x SFP
IP adresa	192.168.88.2

Tabulka 5: Popis HW přepínače Zyxel

Typ	ADSL modem
Model	Linksys X1000
LAN	WAN 1 x 10/100Mbps ,LAN 3 x 10/100Mbps
IP adresa	10.0.0.138

Tabulka 6: Popis HW ADSL modemu

8.2 Instalace aplikace

8.2.1 Konfigurace a natavení Mikrotiku

Protože je v základní nastavení Mikrotiku je rozhraní API vypnuté, bylo zapotřebí rozhraní zapnout, změnit standardní port na 18728 a zpřístupnit API pouze z IP adresy TCP proxy. Tedy z IP adresy 192.168.88.254. Tak jak není v základní nastavení Mikrotiku rozhraní API zapnuté, tak není zapnutá ani Webproxy. Proto jsem ji zapnul, změnil jsem standardní port 8080 na port 8088 a nastavil posílání logů z Webproxy na IP adresu 192.168.88.254 a port 514 Syslogu.

Dalším krokem bylo vložit pravidlo pro blokování internetu. Pravidlo jsem vložil do Firewall/Filter Rules jako první pravidlo. Pravidlo zakazuje veškerý provoz z adresního listu BLOCK skrz Mikrotik na cílových portech 80 a 443.

Pro vytvoření transparentní proxy se přidá pravidlo do NATu, které se vytvoří při registraci učebny. Aby mohla webová aplikace komunikovat s Mikrotikem, musel jsem

vytvořit skupinu WebApp, nastavit ji přístup k Mikrotikm API, nastavit patřičná práva a zařadit do ní nového uživatel WebUser.

```
/ip/ firewall / filter   add chain=forward action=drop protocol=tcp src-address-list=BLOCK dst-
    port=80,443
```

Výpis 5: Příklad pravidla pro blokování internetu

8.2.2 Konfigurace Server

Nejprve jsem provedl instalace rolí DHCP server a webový server IIS7 operačního systému Microsoft Windows Server 2008 Standard R2. Server DNS nebylo potřeba instalovat, protože již byl instalován spolu s řadičem domény.

V dalším kroku jsem provedl instalaci Microsoft SQL Server 2008 R2 Express. I když se nedoporučuje instalovat SQL Server na řadič domény, z důvodu ovlivnění výkonu SQL serveru. Do tabulek CLASSROOMS, USERS a LOGMK se zapisuje jen nepatrně, ale do tabulky PROXLOG se bude zapisovat velké množství záznamů. Pro mé účely bude výkon dostatečný. Pomocí aplikace SQL Server Management Studio a SQL skriptu jsem vytvořil databázovou strukturu. Spolu s databázovou strukturou se vytvoří výchozí uživatel aplikace.

Dále jsem na server nainstaloval platformu Microsoft .NET Framework 4.5, podpůrné knihovny pro běh aplikace Microsoft Web Platform Installer 5.0. Abych mohl snadno nasadit aplikaci, musel jsem nainstalovat na server doplněk Web Deploy v3.0. Díky doplňku jsem mohl přímo z Visual Studia 2012, provést export aplikace na webový server IIS7.

V DHCP serveru jsem vytvořil rezervace pro jednotlivé počítače, tak aby počítače v učebnách měli přiřazené IP adresy za sebou. V učebně CLASSROOM01 je 21 počítačů a v učebně CLASSROOM02 je počítačů 25. Učebna CLASSROOM01 má rozsah 192.168.88.99-192.168.88.119 a učebna CLASSROOM02 192.168.88.120-192.168.88.145. První IP adresa rozsahu učeben je IP adresa učitelské stanice. Aby byla aplikace přístupná z lokální sítě, musel jsem povolit port 80 na bráně firewall. DNS server bude využíván k překladům IP adres, na názvy počítačů.

Nasazení služby ProxyService jsem provedl pomocí aplikace Installutil.exe, která je součástí Visual Studia 2012. Aplikace slouží k registraci služby do operačního systému Windows. Zaregistrované službě ProxyService jsem nastavil potřebné parametry pro spuštění služby.

Parametry pro spuštění služby:

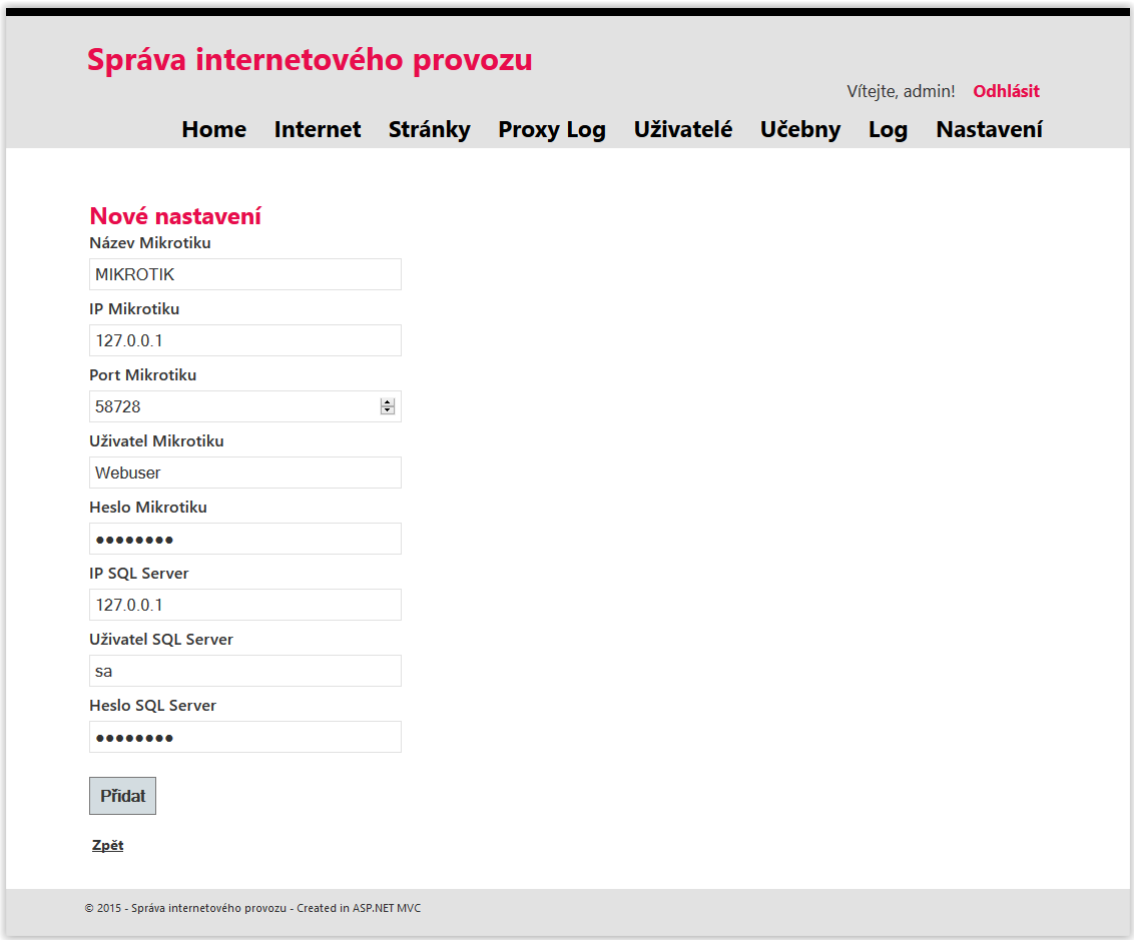
- IP adresa TCP proxy - v mém případě 127.0.0.1
- port TCP proxy 58728
- IP adresu Mikrotiku 192.168.88.1
- port Mikrotik API 18728
- cesta k souboru s výpisem chyb `C:\LOG\errorLog.txt`

Režim spuštění služby jsem nastavil automaticky se zpožděním, aby služba nebrzdila start operačního systému. Služba využívá ke spuštění uživatelský účet doménového administrátora.

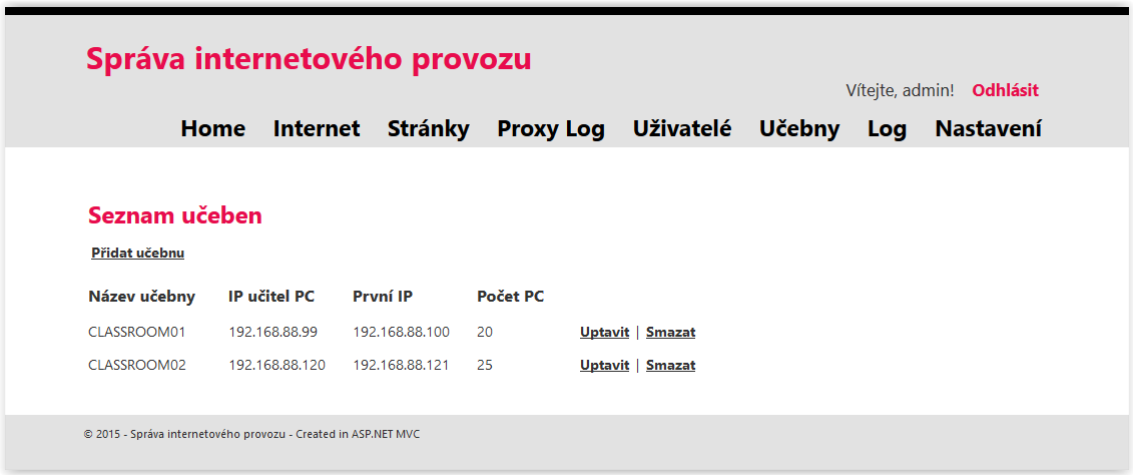
8.2.3 První spuštění

Spuštění aplikace se provádí otevřením URL adresy `http://SERVER01/MVC/Account/Login`.

Přihlášení do aplikace jsem provedl pomocí předem nadefinovaného účtu „admin“. Nejprve jsem v nastavení nadefinoval přístup k Mikrotiku a SQL serveru. Následně jsem provedl registraci obou učeben. A nakonec zbývalo zadat uživatelé webové aplikace.



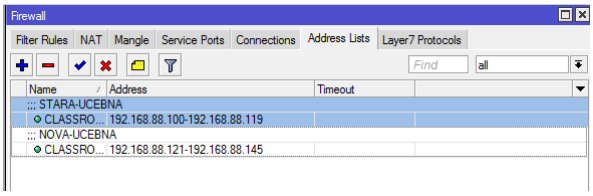
Obrázek 6: Nastavení aplikace



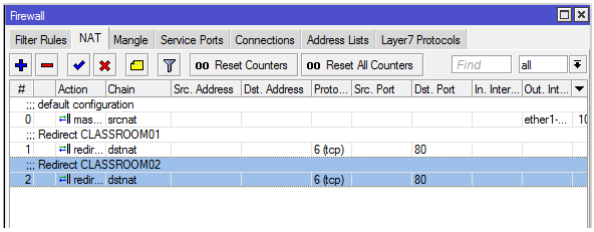
Obrázek 7: Registrace učeben

Obrázek 8: Vytvoření uživatelů

Pro kontrolu, že aplikace je správně nastavená, zkontroloval jsem v Mikrotiku, zda jsou v sekci ADDRESS LIST a NAT zapsány obě učebny.



Obrázek 9: Mikrotik Address List

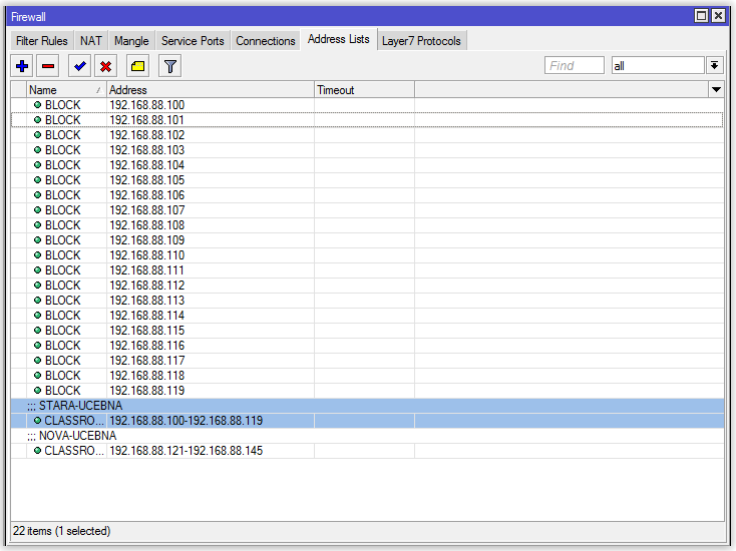


Obrázek 10: Mikrotik NAT

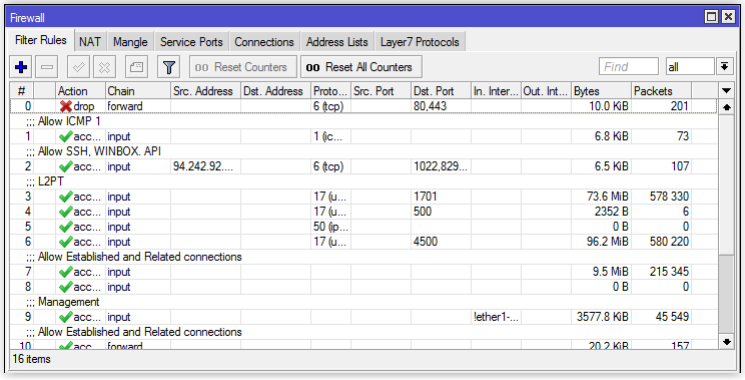
8.3 Testování

K testování jsem použil stanice v počítačové učebně CLASSROOM01. Při testování jsem se do aplikace přihlašoval pod různými uživateli, tak abych ověřil funkčnost uživatelských rolí.

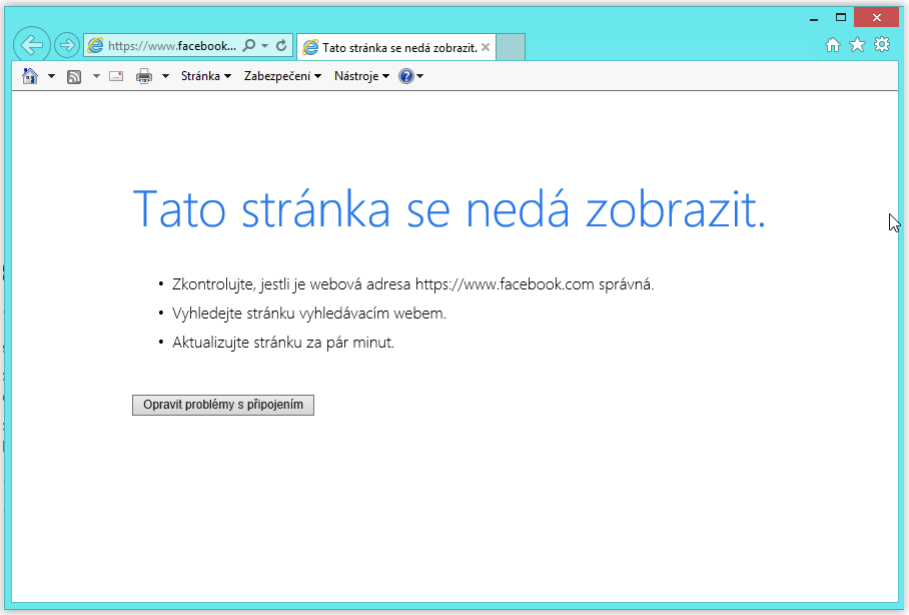
Nejprve jsem otestoval blokování celé učebny. Ve výpisu Address Listu přibyl seznam BLOCK, obsahující žákovské stanice v učebně. Následně začalo pravidlo, pro blokaci internetu zahazovat pakety, směřující k počítačům v seznamu BLOCK.



Obrázek 11: Mikrotik Address List seznam BLOCK

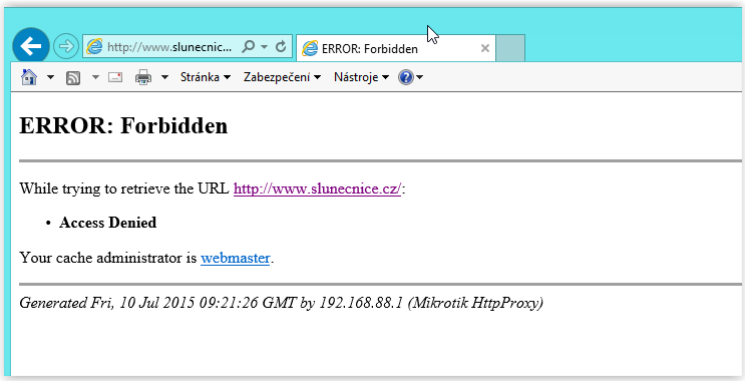


Obrázek 12: Ukázka pravidla při blokaci internetu

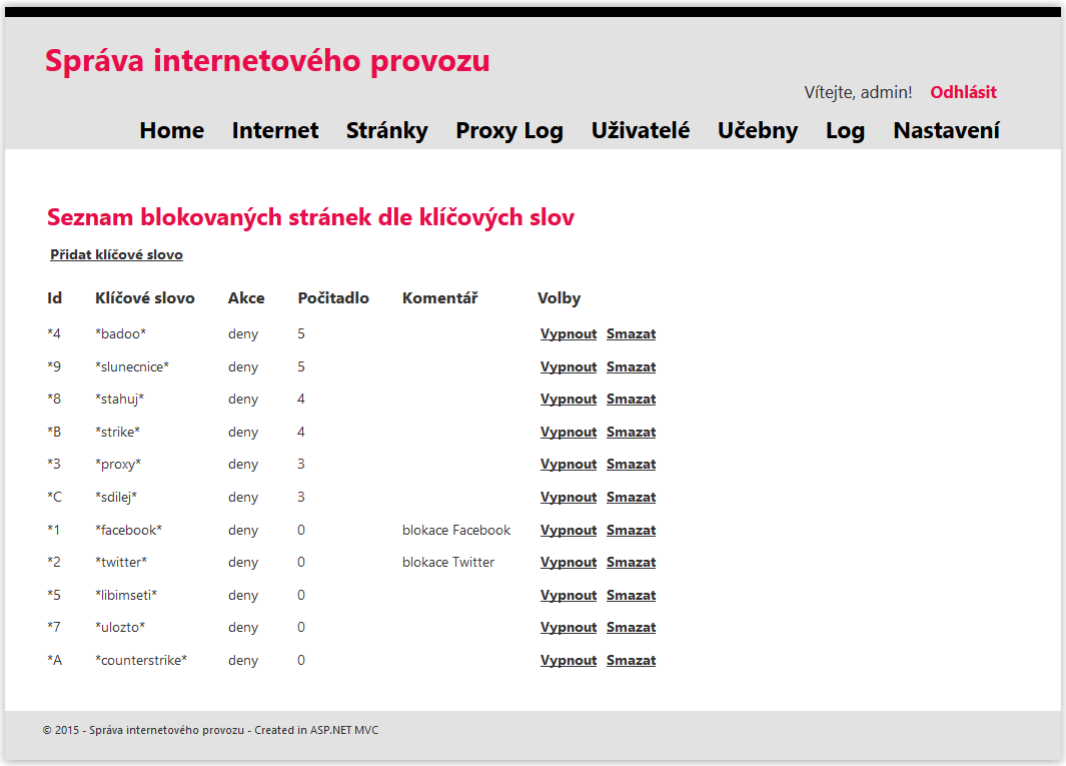


Obrázek 13: Náhled internetového prohlížeče při blokování internetu

K blokování webových stránek na základě klíčových slov, jsem nejprve musel vytvořit seznam klíčových slov. Následně jsem v internetovém prohlížeči zadával webové stránky obsahující klíčová slova. Výsledkem bylo přesměrování na stránku Webproxy, kde jsem byl informován o zablokování stránky.

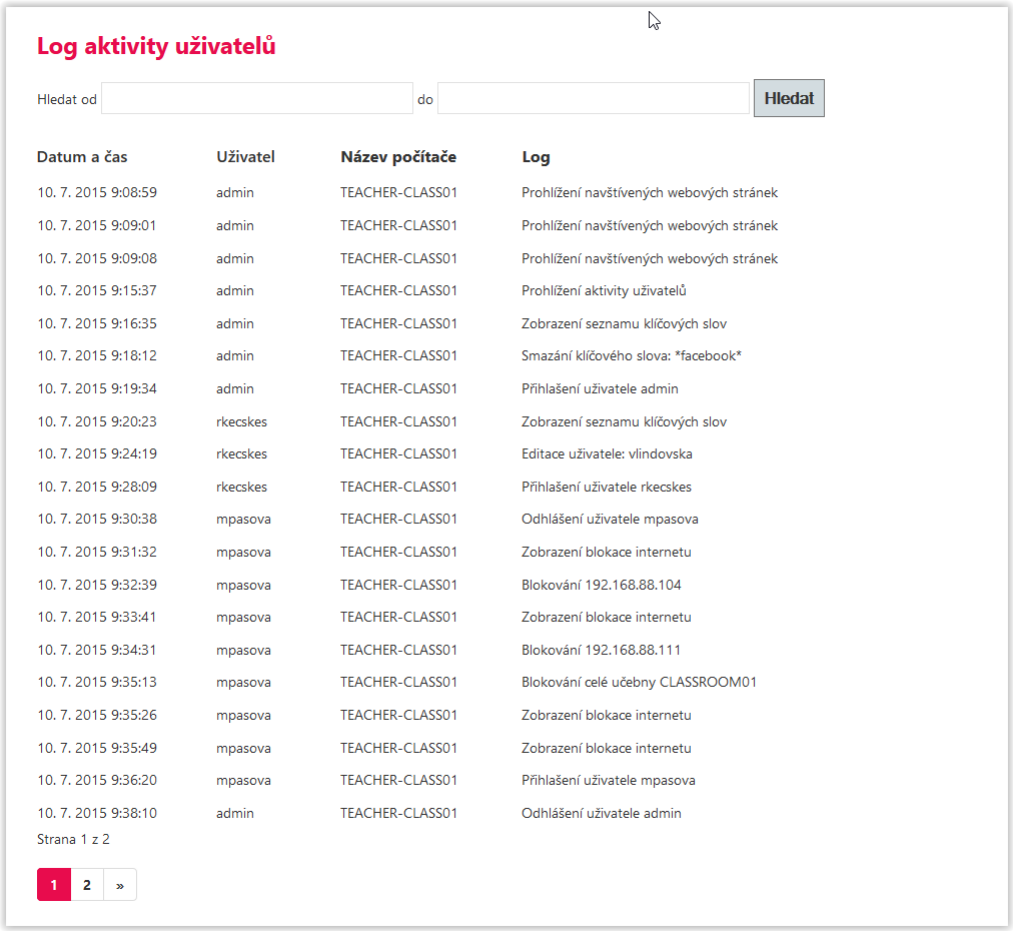


Obrázek 14: Náhled internetového prohlížeče při blokování internetu



Obrázek 15: Ukázka seznamu klíčových slov

V seznamu blokovaných stránek se objevili počty úspěšně zablokovaných pokusů, seřazených sestupně.



Log aktivity uživatelů

Hledat od do Hledat

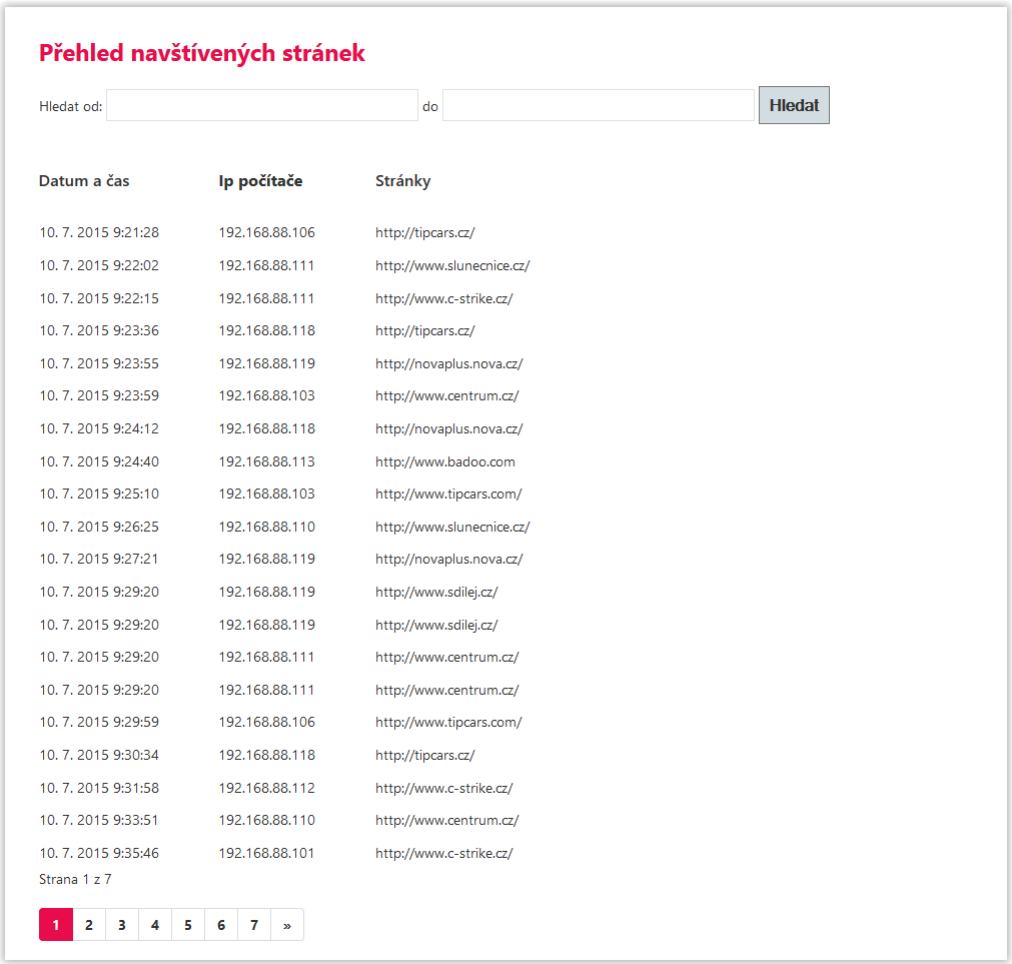
Datum a čas	Uživatel	Název počítače	Log
10. 7. 2015 9:08:59	admin	TEACHER-CLASS01	Prohlížení navštívených webových stránek
10. 7. 2015 9:09:01	admin	TEACHER-CLASS01	Prohlížení navštívených webových stránek
10. 7. 2015 9:09:08	admin	TEACHER-CLASS01	Prohlížení navštívených webových stránek
10. 7. 2015 9:15:37	admin	TEACHER-CLASS01	Prohlížení aktivity uživatelů
10. 7. 2015 9:16:35	admin	TEACHER-CLASS01	Zobrazení seznamu klíčových slov
10. 7. 2015 9:18:12	admin	TEACHER-CLASS01	Smazání klíčového slova: *facebook*
10. 7. 2015 9:19:34	admin	TEACHER-CLASS01	Přihlašení uživatele admin
10. 7. 2015 9:20:23	rkecskes	TEACHER-CLASS01	Zobrazení seznamu klíčových slov
10. 7. 2015 9:24:19	rkecskes	TEACHER-CLASS01	Editace uživatele: vlindovska
10. 7. 2015 9:28:09	rkecskes	TEACHER-CLASS01	Přihlašení uživatele rkecskes
10. 7. 2015 9:30:38	mpasova	TEACHER-CLASS01	Odhlášení uživatele mpasova
10. 7. 2015 9:31:32	mpasova	TEACHER-CLASS01	Zobrazení blokace internetu
10. 7. 2015 9:32:39	mpasova	TEACHER-CLASS01	Blokování 192.168.88.104
10. 7. 2015 9:33:41	mpasova	TEACHER-CLASS01	Zobrazení blokace internetu
10. 7. 2015 9:34:31	mpasova	TEACHER-CLASS01	Blokování 192.168.88.111
10. 7. 2015 9:35:13	mpasova	TEACHER-CLASS01	Blokování celé učebny CLASSROOM01
10. 7. 2015 9:35:26	mpasova	TEACHER-CLASS01	Zobrazení blokace internetu
10. 7. 2015 9:35:49	mpasova	TEACHER-CLASS01	Zobrazení blokace internetu
10. 7. 2015 9:36:20	mpasova	TEACHER-CLASS01	Přihlašení uživatele mpasova
10. 7. 2015 9:38:10	admin	TEACHER-CLASS01	Odhlášení uživatele admin

Strana 1 z 2

1 2 »

Obrázek 16: Ukázka výpisů aktivt uživatelů aplikace

Nakonec zbývá otestovat, zda funguje služba ProxyService, konkrétně Syslog. Důležité je správné nastavení Mikrotiku, tak aby odesílal logy Webproxy na Syslog. Syslog naslouchá na portu 514, pokud přijme logy z Webproxy, následně je zapíše do databáze. K ověření, stačí zobrazit stránku PROXY LOG. Samozřejmě můžeme ověřit funkčnost v SQL Server Management Studiu, zobrazením obsahu tabulky PROXYLOG.



Obrázek 17: Ukázka výpisů navštívených webových stránek

K ověření funkčnosti TCP proxy by mělo stačit to, že aplikace komunikuje s Mikrotikem. Pokud by TCP proxy nefungovala, nefungovala by ani aplikace.

8.4 Výsledky testování

Testování probíhalo na počítačové učebně CLASSROPOM01. Při testování se podařilo ověřit veškerou funkčnost aplikace. Po bližším zkoumání jsem objevil občasné duplicitní záznamy v PROXY LOGu. Duplicita byla zřejmě způsobena nedostatkem výkonu ze strany Mikrotiku, při zpracovávání výpisů logů Webproxy. Proto bych se zaměřil na zestručnění a omezení logů z Webproxy.

8.5 Další vývoj aplikace

Z hlediska bezpečnosti bych využil šifrovaného přenosu mezi webovou aplikací a Mikrotikem. Mikrotik v dnešní době disponuje šifrovaným přenosem Mikrotik API. Nasazení aplikace na webový server s podporou šifrované komunikace.

Zlepšit monitorování síťové aktivity dopracováním TCP proxy, o možnost ukládat podrobnější výpis komunikace do databáze. Protože dnes převážná většina webových stránek již využívá šifrované komunikace, tak blokování webových stránek řešené pomoci Webproxy je nedostačující. Proto je třeba zvážit nasazení MITM proxy nebo využít funkce Layer7, spolu s pravidlem firewallu.

Z hlediska uživatelského prostředí, bylo možné připravit lepší ovládán blokování internetu.

9 Závěr

Cílem bakalářské práce bylo navrhnout a implementovat rozhraní pro správu routeru na platformě Mikrotik. Stanovených cílů bylo dosaženo ve všech bodech. Rozhraní bylo nasazeno jako webová aplikace, byla vytvořena uživatelská příručka a rozhraní se podařilo otestovat v reálném provozu na síťové architektuře popsané v kapitole 4. Komunikace mezi webovou aplikací a routerem na platformě Mikrotik byla realizována pomocí TCP proxy. Systém se při testování jevil stabilně, ale v oblasti ukládání výpisů logů Webproxy do databáze, pomocí Syslogu vykazoval občasné uložení duplicitních záznamů. Duplicitu zřejmě způsobuje nedostatek hardwarových zdrojů Mikrotik. Pro složitější síťové architektury, bych určitě doporučil výkonnější verzi Mikrotiku. Seznámení se s metodikou návrhu, implementací a nasazením v reálném provozu, mělo pro mě velký přínos. Získané poznatky a zkušenosti měli do velké míry vliv na rozvoj mých znalostí v této oblasti. Věřím, že využití této bakalářské práce, bude mít nejen přínos pro mě, ale také pro zaměstnavatele. V praxi bylo ověřeno, že aplikace je funkční a je svou otevřenou architekturou připravena k dalšímu rozšíření v oblasti správy internetového provozu.

Roman Bednář

10 Reference

[1] STREBE, Matthew a Charles PERKINS. *Firewally a proxy-servery*. Vyd. 1. Brno: Computer Press, 2003, xxi, 450 s. ISBN 80-722-6983-6.

[2] MACDONALD, Matthew, Adam FREEMAN a Mario SZPUSZTA. *ASP.NET 4 a C# 2010: tvorba dynamických stránek profesionálně*. Vyd. 1. Překlad Jan Pokorný. Brno: Zoner Press, 2011, 880 s. Encyklopedie Zoner Press. ISBN 978-80-7413-131-8.

[3] Manual: Mikrotik API. [online]. [cit. 2013-09-31]. Dostupné z: http://wiki.mikrotik.com/wiki/Manual:API

[4] Manual: Mikrotik. [online]. [cit. 2013-09-31]. Dostupné z: http://wiki.mikrotik.com/wiki/

[5] MVC : Official Microsoft Site *Getting Started with ASP.NET MVC 3* http://www.asp.net/mvc [cit. 15.7.2014]

[6] METASTRUCT. Multithreaded, Customizable SysLog Server - C#. [online]. [cit. 2014-07-30]. Dostupné z: `http://www.codeproject.com/script/Articles/ArticleVersion.aspx?aid=441233&av=633101`

[7] GARCIA, Bruno. Simple TCP Forwarder in C#. [online]. [cit. 2014-07-30]. Dostupné z: `http://blog.brunogarcia.com/2012_10_01_archive.html`

[8] PASEKA, Radim. *Omezování nežádoucího provozu na bezdrátových sítích*, VŠB-TU 2004 http://www.cs.vsb.cz/grygarek/SPS/materialy/Paseka_DP/dp.pdf

[9] Intro to ASP.NET Web Programming Razor Syntax : Official Microsoft Site *Introduction to ASP.NET Web Programming Using the Razor Syntax.* http://www.asp.net/web-pages/tutorials/basics/2-introduction-to-asp-net-web-programming-using-the-razor-syntax [cit. 20.7.2014] *The LaTeX companion,* New York: Addison, 1994.

[10] Lamport, Leslie, *LaTeX: a document preparation system: user's guide and reference manual*, New York: Addison-Wesley Pub. Co., 1994.

[11] MAREK, Vlastimil. Něco v síti: fejetony, které vycházely od roku 1997 na internetu na adrese http://svet.namodro.cz [online]. [cit. 2015-07-12]. DOI:`http://www.mediapodlupou.cz/lekce/deti-a-mladez-v-kyberprostoru`

[12] Co je Internet a jak funguje? [online]. 26.01.2010. 2010 [cit. 2015-07-12]. Dostupné z: `http://datacentrum.wedos.com/a/17/co-je-internet-jak-funguje.html`

[13] *PC Control* [online]. [cit. 2015-07-16]. Dostupné z: `http://www.pc-control.cz`

[14] *Správce učebny - SODATSW spol. s r.o.* [online]. [cit. 2015-07-16]. Dostupné z: `http://www.sodatsw.cz/spravce-ucebny`

[15] *Classroom Management, Monitoring, and Corporate Training Software | LanSchool* [online]. [cit. 2015-07-16]. Dostupné z: http://www.lanschool.com

[16] *Kerio Control | Kerio Technologies* [online]. [cit. 2015-07-16]. Dostupné z: http://www.kerio.cz/products/kerio-control

[17] *Manual:TOC - MikroTik Wiki* [online]. [cit. 2015-07-16]. Dostupné z: http://wiki.mikrotik.com/wiki/Manual:TOC

[18] API in C Sharp. [online]. [cit. 2015-07-19]. Dostupné z: http://wiki.mikrotik.com/wiki/API_in_C_Sharp

[19] Bc. R. Taraba, *Ovládání Mikrotik RouterOS z mobilního zařízení*, diplomová práce, VŠB-TU FEI, Ostrava, 2011

[20] *STANOVISKO č. 2/2009 - Úřad pro ochranu osobních údajů* [online]. [cit. 2015-07-19]. Dostupné z:https://www.uoou.cz/files/stanovisko_2009_2.pdf

[21] Oficiální stránky výrobce Linksys. [online]. [cit. 2015-07-19]. Dostupné z: http://www.linksys.com/fr/p/P-X1000/

[22] *Man in the middle* [online]. [cit. 2015-07-19]. Dostupné z: https://cs.wikipedia.org/wiki/Man_in_the_middle

11 Přílohy

11.1 Příloha na CD/DVD

- Elektronická kopie bakalářské práce
- Praktická část bakalářské práce
- Skripty SQL

11.2 Uživatelská příručka

11.2.1 Přihlášení

K přihlášení do sytému použijeme uživatelské jméno a heslo. Pokud se jedná o prvotní přihlášení do systému, použijeme předem nadefinovaný administrátorský přístup, vytvořený spolu s databázovou strukturou. V SQL skriptu je nadefinovaný uživatel „admin“ s heslem „P@ssword“.

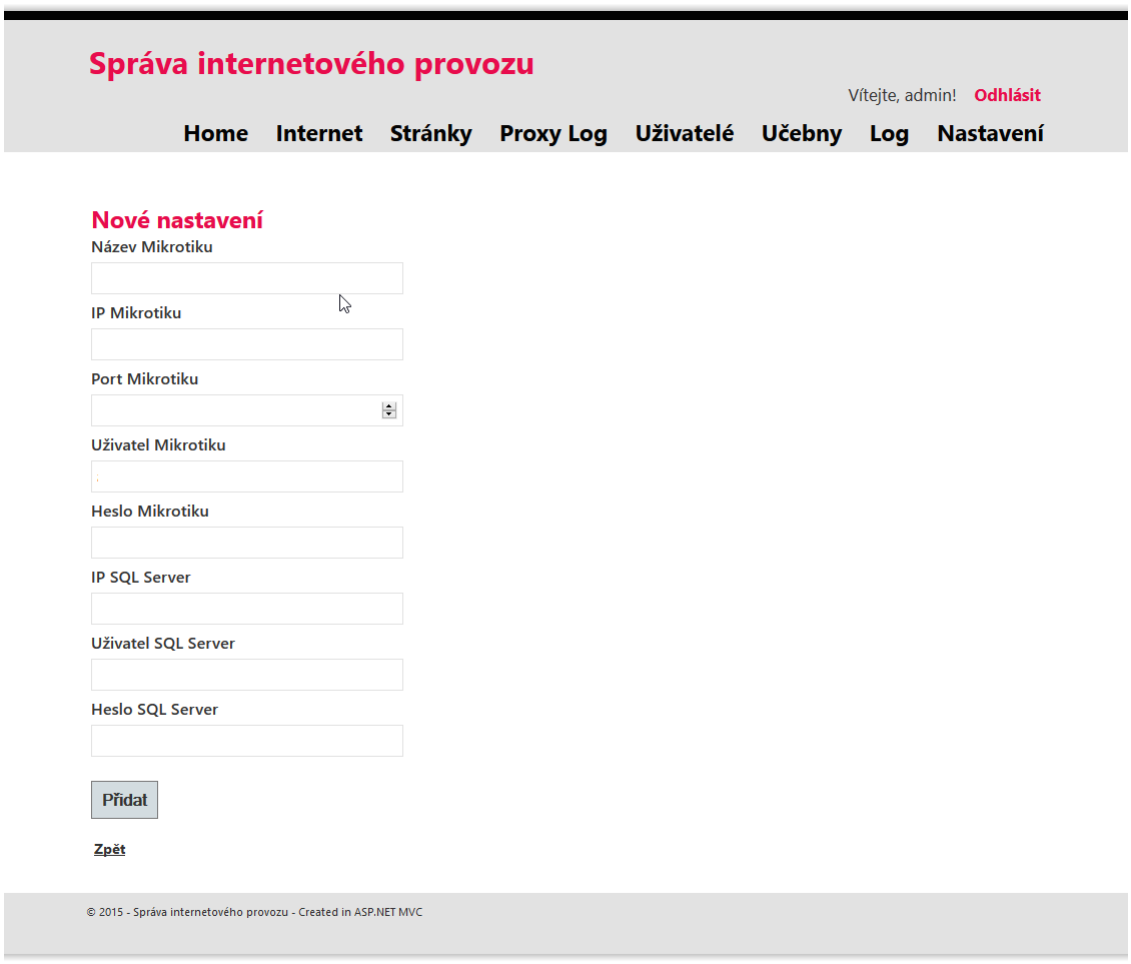
Obrázek 18: Stránka pro přihlášení do aplikace

Po přihlášení se zobrazí úvodní stránka s základním popisem aplikace. Položky nabídky závisí na roli uživatele. Na obr. 2 je zobrazena nabídka uživatele s rolí Administrators. Users mají v nabídce Home, Internet, Stránky a Proxy Log. Power users mají navíc položky Uživatel, Učebny a Log.

Obrázek 19: Úvodní stránka aplikace

11.2.2 Nastavení aplikace (Nastavení)

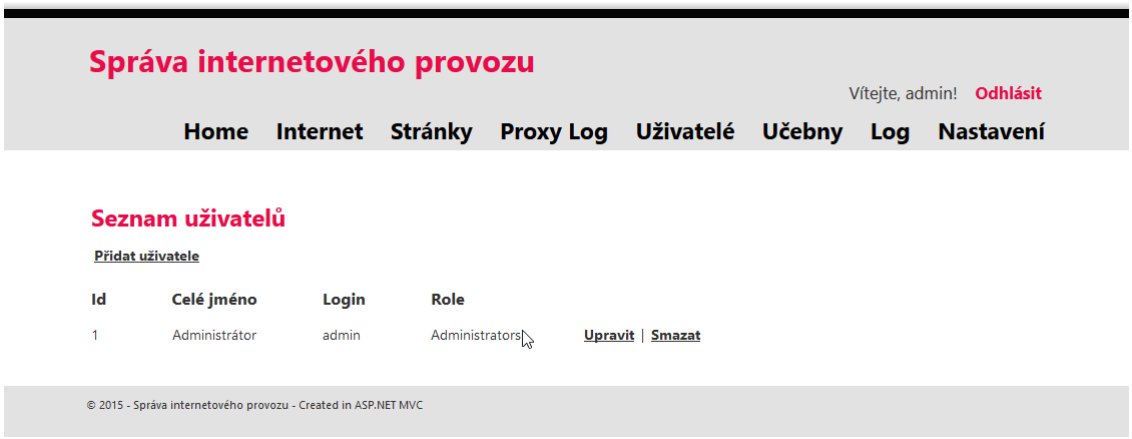
Pro správnou funkčnost aplikace je zapotřebí nastavení přístupu k Mikrotiku a SQL databázi. U Mikrotiku nastavíme IP adresu a port TCP Proxy, přihlašovací údaje uživatele, který má k rozhraní Mikrotik API přístup. Dále budeme potřebovat zadat přístup k databázi, kde zadáme IP adresu SQL serveru, uživatelské jméno a heslo.



Obrázek 20: Formulář pro zadání nastavení aplikace

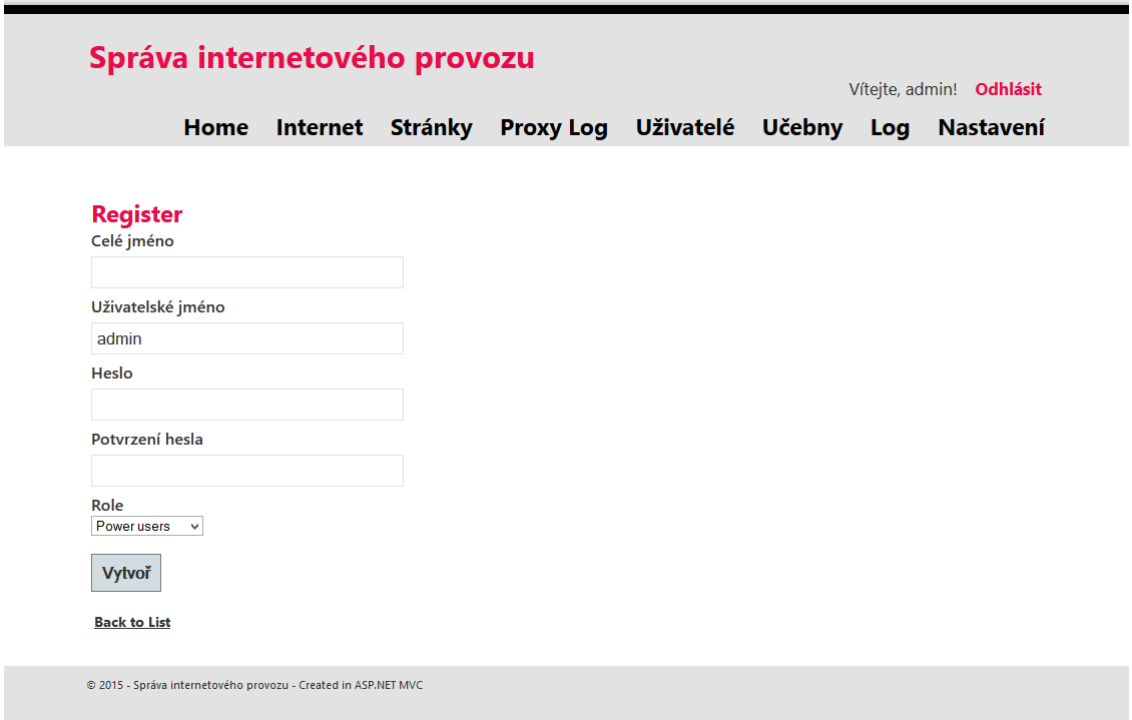
11.2.3 Správa uživatelů (Uživatelé)

Jako výchozí uživatel je *„admin“* s administrátorským přístupem. Pro přidání dalších uživatelů použijeme odkaz *„Přidat uživatele“*.



Obrázek 21: Seznam uživatelů aplikace

U nového uživatele zadáme celé jméno, uživatelské jméno, které slouží zároveň jako login, heslo a nakonec zvolíme, jakou bude mít nový uživatel přiřazenou roli.



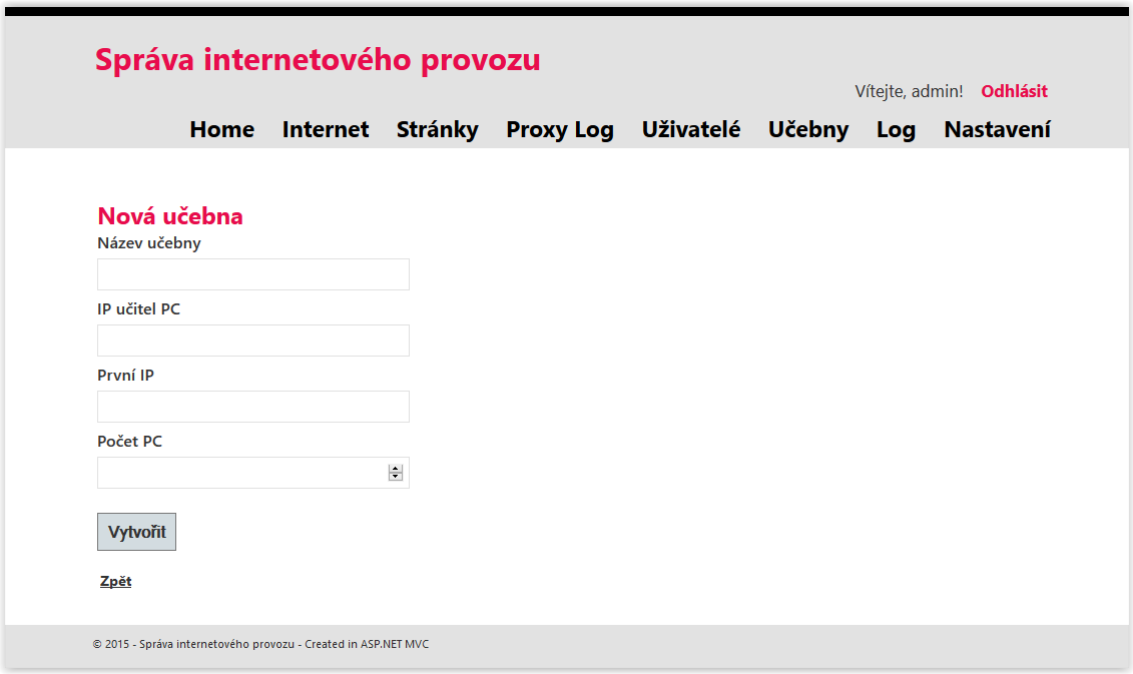
Obrázek 22: Formulář pro vytvoření nového uživatele

11.2.4 Registrace učebny (Učebny)

Přidání učebny provedeme kliknutím na záložku *„Učebny“* a na odkaz *„Přidat učebnu“*.

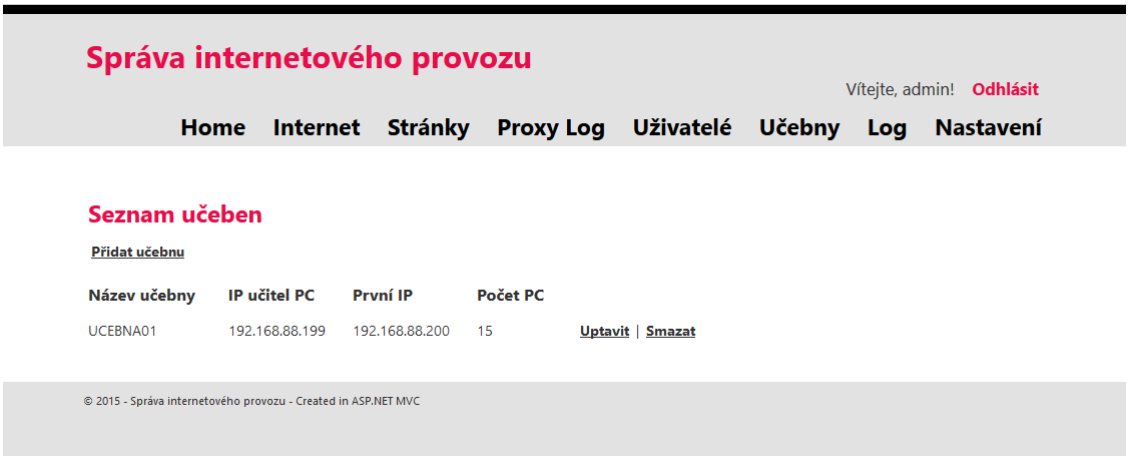
Obrázek 23: Výpis registrovaných učeben

Zadáním názvu učebny, IP adresy učitelské stanice, IP adresy prvního žákovského počítače a počtu stanic v celé učebně, zaregistrujeme novou učebnu.



Obrázek 24: Formulář pro registraci nové učebny

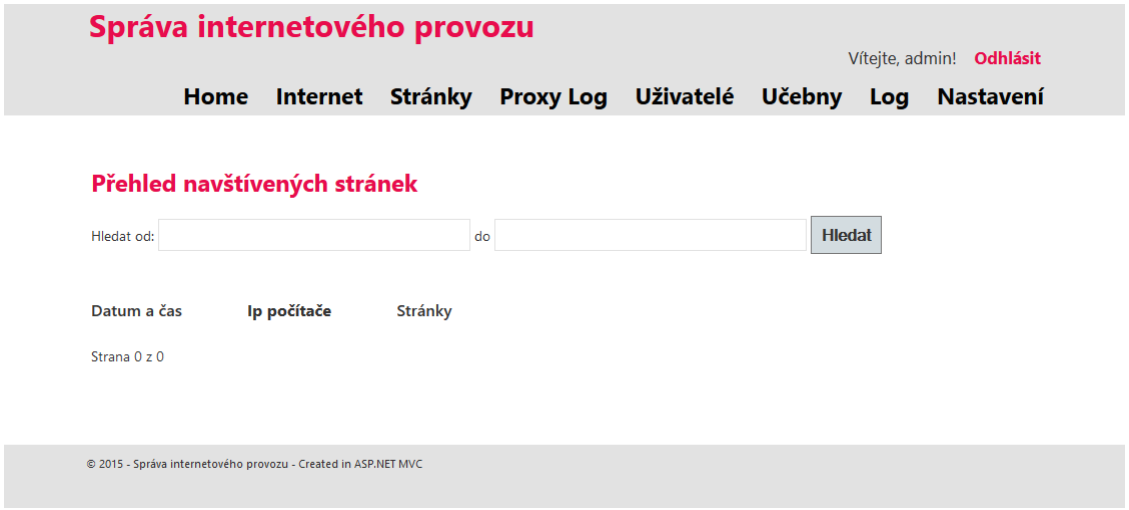
Uživatelé s rolí „Power users“ nebo „Administrators“ mohou učebny upravovat. Mazání učeben je povoleno pouze Administrators.



Obrázek 25: Výpis registrovaných učeben

11.2.5 Výpis navštívených stránek (Proxy Log)

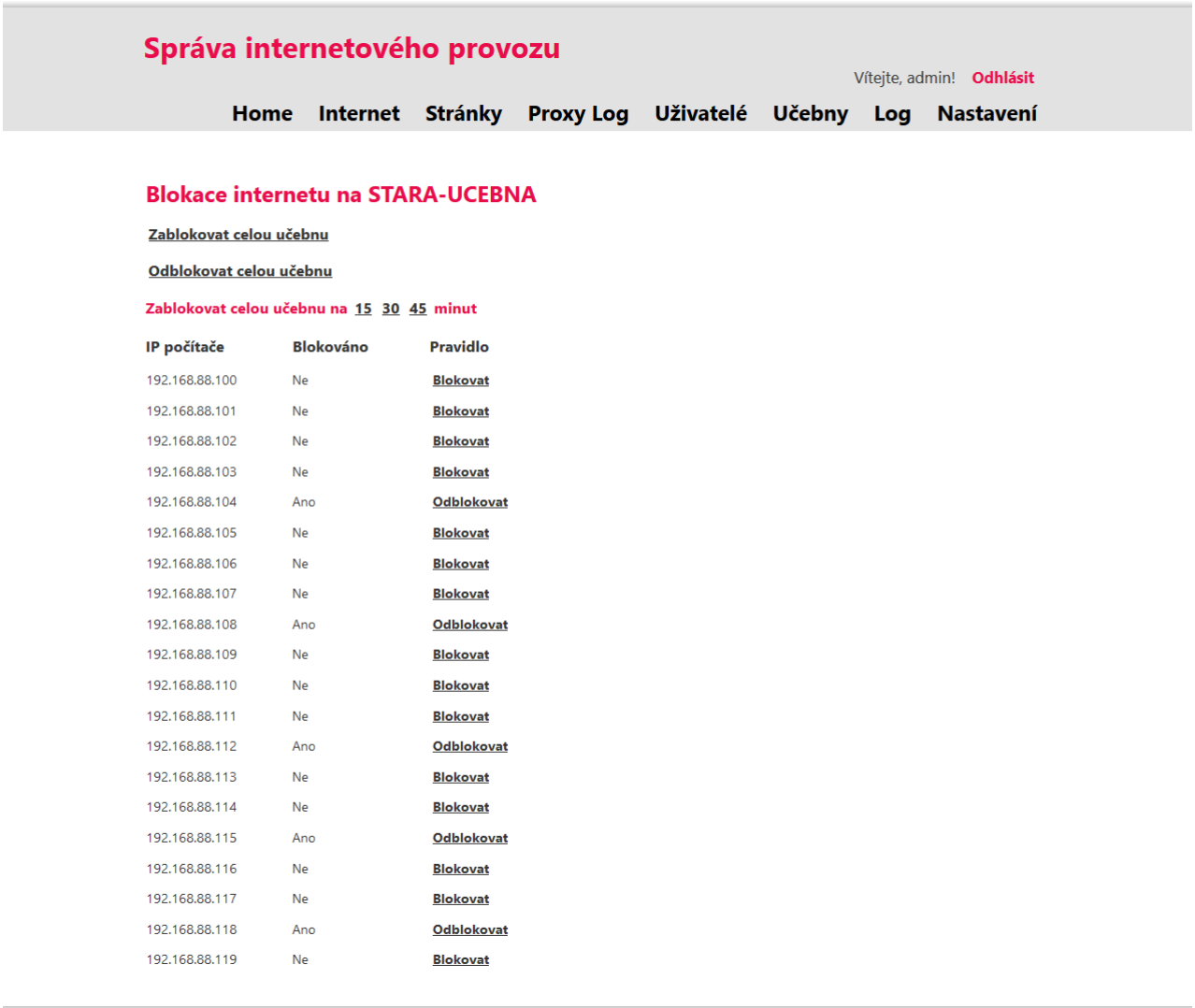
Proxy Log vypíše seznam navštívených stránek žáky, s výchozím řazením od nejnovějšího záznamu. Řazení je možno měnit kliknutím na popisek *„Datum a čas“* nebo *„Stránky“*. Výpis je možno filtrovat podle data a času. Pro přehlednější procházení výpisu je výpis stránkován.



Obrázek 26: Výpis navštívených stránek

11.2.6 Blokování internetu (Internet)

Zobrazení stránky Internet, nabídne výpis stavu blokování žákovských stanic na dané učebně.



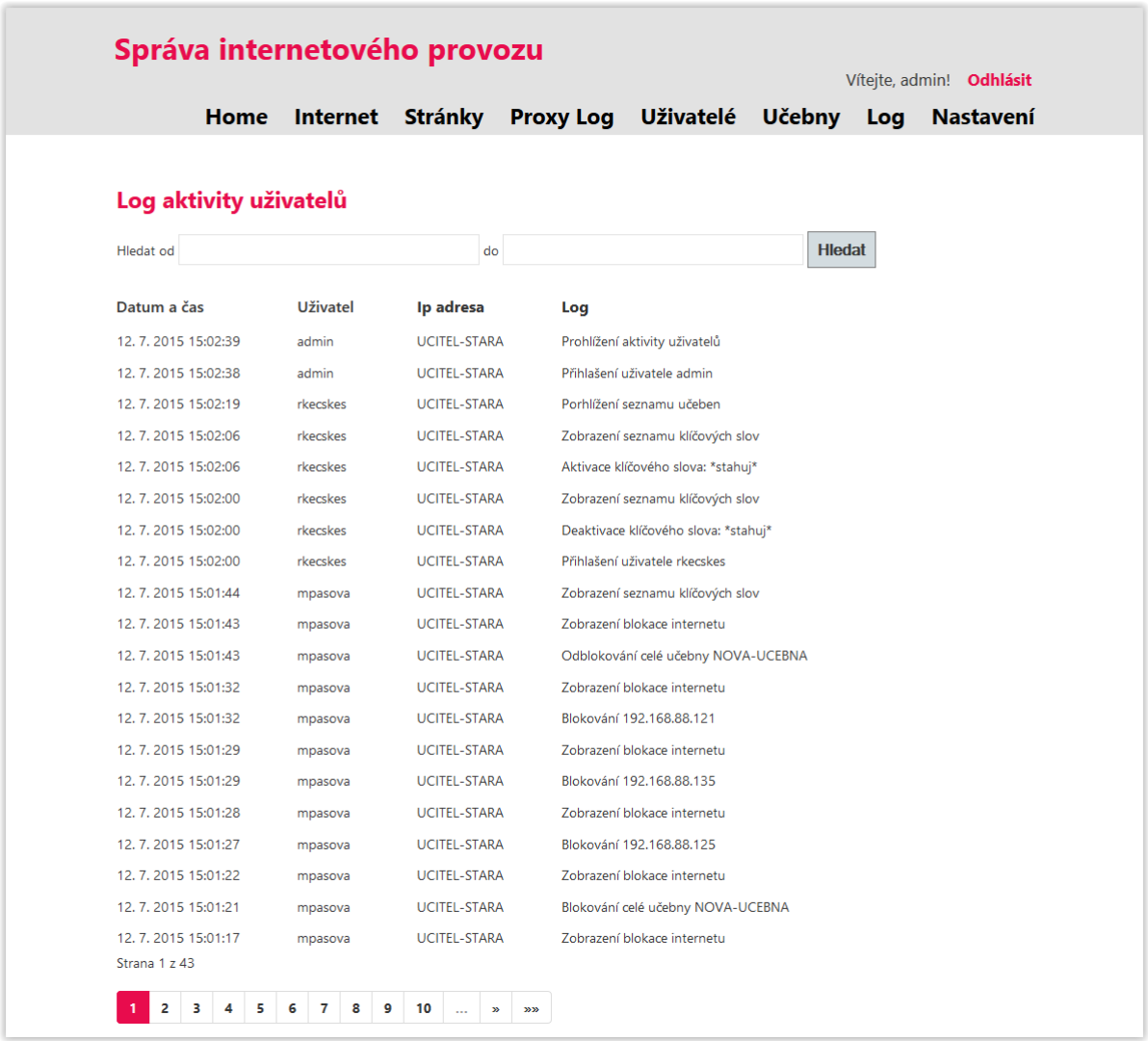
Obrázek 27: Stránka správy přístupu na internet

Volbou zablokovat/odblokovat celou učebnu nastavíme blokování/odblokování internetu všem žákovským stanicím na učebně. Pro nastavení časového blokování internetu všem žákovským stanicím vybereme z možností 15, 30 nebo 45 minut.

Pro blokovaní/odblokování internetu pro jednotlivé žákovské stanice zvolíme u požadované stanice volbu blokovat/odblokovat.

11.2.7 Prohlížení aktivity uživatelů (Log)

Log vypíše seznam aktivit uživatelů aplikace, s výchozím řazením od nejnovějšího záznamu. Řazení je možno měnit kliknutím na popisek *„Datum a čas“* nebo *„Stránky“*. Výpis je možno filtrovat podle data a času. Pro přehlednější procházení výpisu je výpis stránkován.



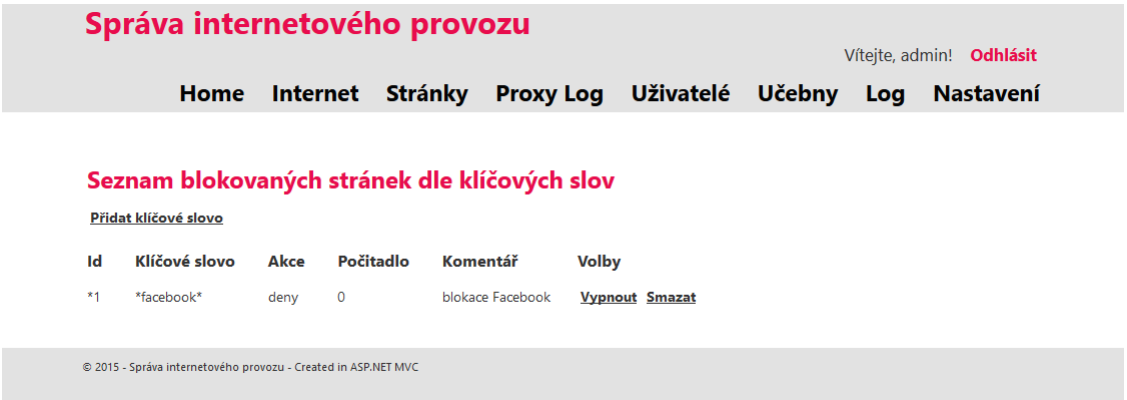
Obrázek 28: Výpis aktivit uživatelů

11.2.8 Správa webových stránek (Stránky)

V záložce „*Stránky*“ je k dispozici seznam blokovaných stránek. Seznam je řazen dle počtu navštívených stránek, tedy dle sloupce „*Počitadlo*“.

Obrázek 29: Formulář zadání nového klíčového slova nebo URL adresy

Blokované stránky se zadávají buď jako klíčové slovo ve tvaru „*slovo*“ nebo jako konkrétní URL adresa, např. „`www.facebook.com`“. K jednotlivým záznamům je možno vkládat komentáře. Při vkládání nových záznamů je nutné zvolit jeden z režimů, zakázat nebo povolit. Pokud je zvoleno zakázat, stránky obsahující klíčové slovo nebo konkrétní URL adresu jsou blokovány. V případě volby povolit, se pouze zaznamenávají přístupy.



Obrázek 30: Stránka správy blokovaných stránek

Správa blokovaných stránek umožňuje jednoduché povolení nebo zakázání klíčových slov, URL adres, bez nutnosti vytvářet nebo mazat záznamy. Volbu smazat má k dispozici uživatel s rolí *„Administrator“* a *„Power users“*.

11.2.9 Odhlášení

Odhlášením dojde k ukončení práce s aplikací a přesměrování na stránku s přihlášením.